

Contents

EXECUTIVE SUMMARY	1
USECASE BACKGROUND.....	1
OBJECTIVE	2
BUILDING THE ERP CLOUD PO EVENT USE CASE.....	2
Prerequisites	2
Import VBCS Web Application	3
Creating Connections (Flow1).....	4
Creating REST Connection.....	5
Creating the PO Event Integration (Flow1)	7
Activate Integration (Flow1)	31
Testing the ERP Event Flow Integration (Flow1)	32
Creating Connections (Flow2).....	38
Creating the PO Proxy Integration (Flow2)	38
Activate Integration (Flow2)	47
Invoke ERP Cloud Service from VBCS Web App	48
Running the VBCS Web App	54

EXECUTIVE SUMMARY

Oracle Integration provides native connectivity to Oracle and non-Oracle Software as a Service (SaaS) and On-premises applications, such as Oracle ERP Cloud, Oracle Service Cloud, HCM Cloud, Salesforce.com, Workday, EBS, SAP, NetSuite and so on. Oracle Integration adapters simplify connectivity by handling the underlying complexities of connecting to applications using industry-wide best practices. You only need to create a connection that provides minimal connectivity information for each system.

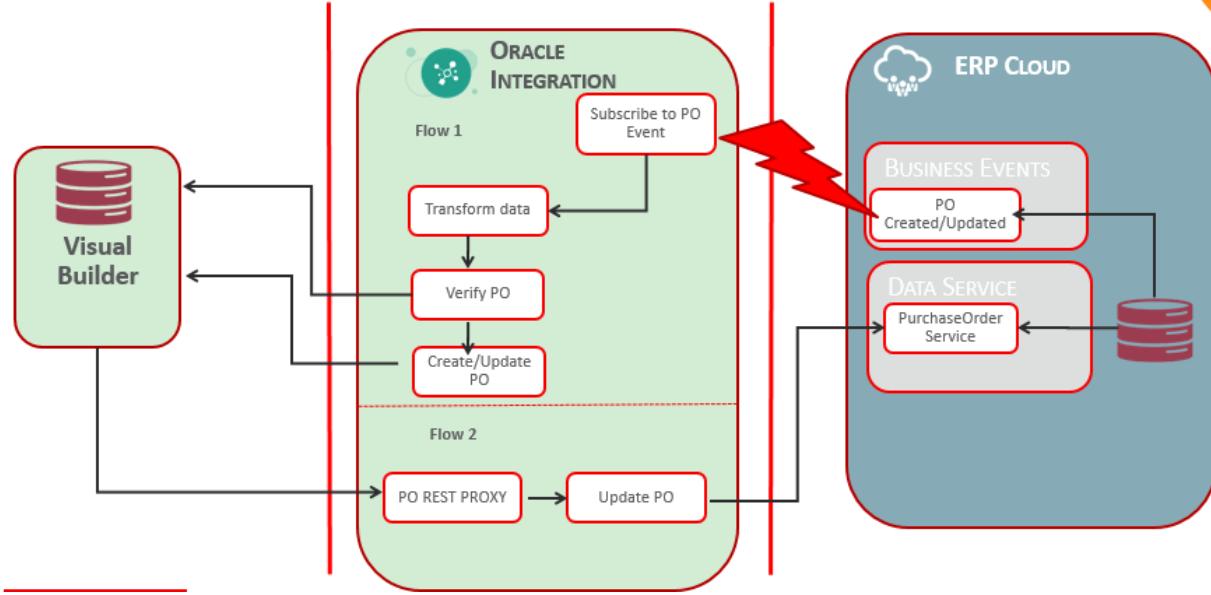
USECASE BACKGROUND

This use case explores the use of Oracle Integration to subscribe to Oracle ERP Cloud Events and push the relevant event information to downstream systems. As part of the lab you would be building the below use case scenario.

- a. User creates a Purchase Order(PO) in ERP Cloud and a PO event is raised (Flow1)
- b. Oracle Integration subscribes to the PO event, transforms and pushes the relevant information to a custom table (VBCS) (Flow1)
- c. User attaches Letter of Credit (LOC) information to the PO record from the VBCS Web App. (Flow2)
- d. PO Record in ERP Cloud is updated with the LOC information in ERP cloud (Flow2)

The following diagram illustrates the proposed interaction between the systems involved in this use case. There 2 flows to complete this use case.

Near Real-Time Synchronization of Purchase Order



OBJECTIVE

This document walks you through the steps needed to replicate this use case in your environment

BUILDING THE ERP CLOUD PO EVENT USE CASE

This section works through the steps that are required to build the integration from scratch.

Prerequisites

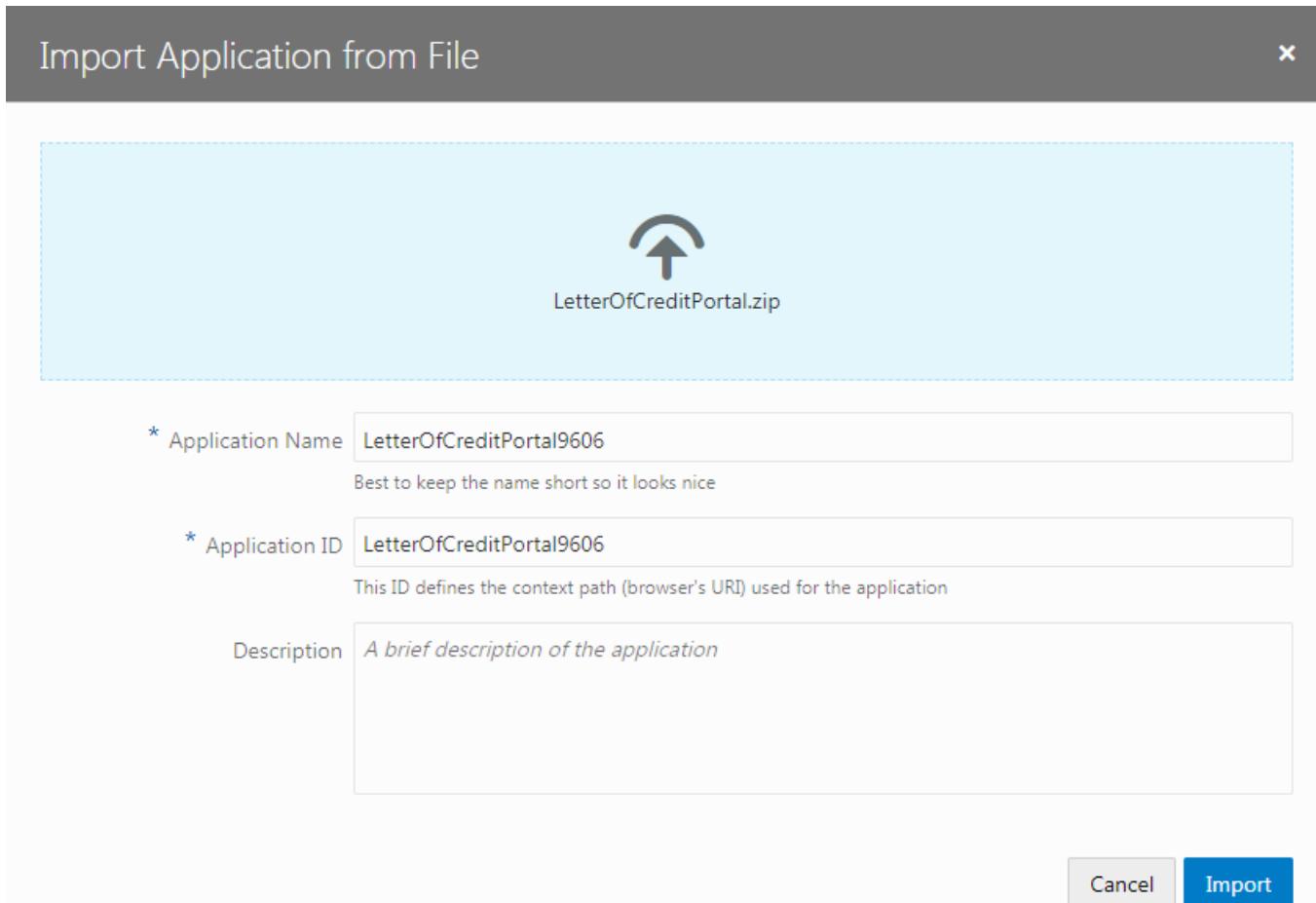
You will need access to the following applications and artifacts:

- Oracle Integration (OIC)
- ERP Cloud R12+
- VBCS App to import (LetterOfCreditPortal.zip)

Import VBCS Web Application

1. Please use Google Chrome browser
2. Login to Oracle Integration homepage and Click on Visual Builder
3. Select Import -> Application from file and Upload the “LetterOfCreditPortal.zip” from the Lab artifacts

Note: Since there are multiple attendees sharing a single environment, please use the `<ClassID><StudentID>` information (will be provided to you by the instructor) as a suffix to your web application
E.g. If your `<ClassID>` is 96 and your `<StudentID>` is 06, then you will name your web application as **LetterOfCreditPortal9606**
4. Change the ApplicationId also to the same name (**LetterOfCreditPortal9606**) to avoid conflicts and Select **Import**



Import Application from File

LetterOfCreditPortal.zip

* Application Name
Best to keep the name short so it looks nice

* Application ID
This ID defines the context path (browser's URI) used for the application

Description

Cancel **Import**

5. Confirm that your application is imported into VBCS. Open the Imported application and you would notice that a few pages and business objects relevant to the use case is pre created. We will use this web application in the later part of the lab.

Select the imported Web Application and you will observe a few pages are pre created

The screenshot shows the Oracle Visual Builder interface. On the left, there's a sidebar with a 'Web Apps' section. Under 'Web Apps', there's a tree view with 'locapp' expanded, showing 'flows +', 'main +', and 'resources'. Under 'main +', four pages are listed: 'editpage', 'iclistpage', 'main-page', and 'polistpage'. The 'editpage', 'iclistpage', and 'main-page' items are highlighted with a red box. The 'Welcome' tab is selected at the top right.

6. Couple of Business Objects (PO and LOC) pre created.
7. Click the PO Business Object, copy the Data endpoint as highlighted below, and save it for later use.

The screenshot shows the Oracle Visual Builder interface for Business Objects. The title bar says 'Business Objects'. The navigation bar includes tabs for 'Welcome', 'editpage', 'PO', 'POPROXY', 'editpage/ChangeOrderAction', 'LOC', and 'Data'. The 'Endpoints' tab is currently selected. In the main area, there's a 'Resource APIs' section with 'Metadata' and 'Data' tabs. The 'Data' tab is selected and has a red box around it. Below the tabs, there are sections for 'Development', 'Staging', and 'Live'. The 'Development' section contains a URL: 'https://octraining2-oicpm.integration.ocp.oraclecloud.com/ic/builder/design/LetterOfCreditPortal9606/1.0/r'. This URL is also highlighted with a red box.

Creating Connections (Flow1)

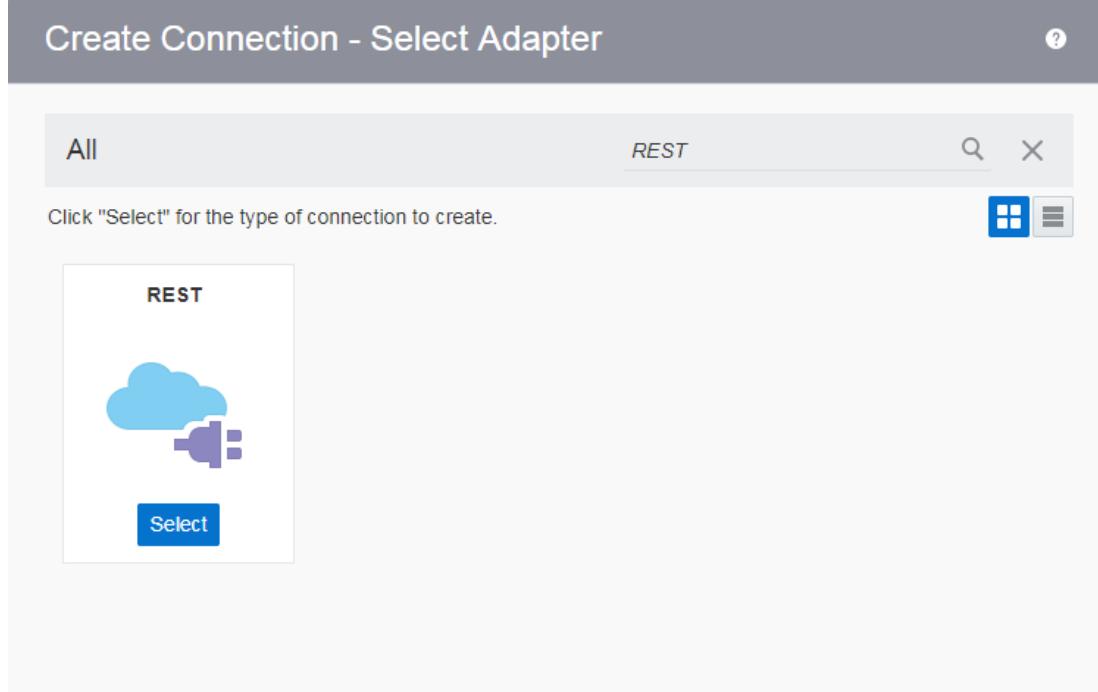
Integrating Your ERP with Oracle Integration

The following Connections have been created and configured. You will be using these connections for creating Integration flows

Connection Name	Connection Type
ERP Conn 96 06	ERP Adapter

Creating REST Connection

1. On the Oracle Integration home page, click Integrations
2. Select the Designer->Connections tab
3. Click on Create and search for REST Adapter by providing the search keyword “REST” and Select Create



4. Provide following details and click on Create

Field	Enter
Name	VBCS REST Con <ClassId><StudentId> Eg: VBCS REST Con 96 06
Identifier	Nothing to enter. The value is automatically generated from the integration name: VBCS_REST_CON_<ClassId>_<StudentId> Eg: VBCS_REST_CON_96_06
Role	Invoke

Create New Connection

Enter information that describes the connection. Use a meaningful name and description to help others find your connection when they create their own integrations. The Identifier must be unique and can be set only when the connection is created.

* Name	VBCS REST Con 96 06
* Identifier	VBCS_REST_CON_96_06
Role	Invoke
Description	Enter a brief description...

Create **Cancel**

5. Configure Connectivity and Security Property Values as per below table and Click "OK"

Field	Enter
Configure Connectivity	
Connection type	REST API Base URL
Connection URL	Use the Data endpoint url copied from "Import VBCS App" Section" -> Step 7 Provide the endpoint till "/data " Ex: https://<oichost>/ic/builder/design/LetterOfCreditPortal9606/1.0/resources/data
Configure Security	
Security Policy	Basic Authentication
Username	Use the same credentials as used for login into Oracle Integration console
Password	Use the same credentials as used for login into Oracle Integration console

Connection Properties

Connection to specify information to connect to your application/endpoint and process requests.

Connection Type

REST API Base URL

TLS Version **OPTIONAL**

< Please select an item from the list >

Connection URL

https://octraining1-oicpm.integration.ocp.oraclecloud.com/ic/builder/design/LetterOfCreditPortal0904/1.0/resources/data

Enable two way SSL for outbound connections **OPTIONAL**

< Please select an item from the list >

Identity keystore alias name **OPTIONAL**

Name of alias to use for establishing identity during two way SSL communication

Security

Security to specify the login credentials to access your application/endpoint.

Security Policy

Basic Authentication

Username	oic.demo
Password

Save and Test the connection

Integrating Your ERP with Oracle Integration

The screenshot shows the Oracle Product Management interface. At the top, there's a green status bar with a checkmark icon and the text "Connection VBCS REST Con 96 06 was tested successfully." Below this is a blue header bar with the title "VBCS REST Con 96 06". On the right side of the header are buttons for "Test", "Close", and "Save". Underneath the header is a grey navigation bar with the text "REST Connection" and a progress bar indicating "100%". To the right of the progress bar are the text "Last Saved: Just now" and a three-dot menu icon.

Creating the PO Event Integration (Flow1)

1. On the Oracle Integration home page, click **Integrations**.
2. On the Integrations page, click **Create**. The Create Integration - Select a Style/Pattern dialog is displayed.
3. Select **App Driven Orchestration** type of integration. The **Create New** Integration dialog is displayed.
4. Enter the following information:

Field Element	Value
What do you want to call your integration?	ERP PO Event VBCS <ClassId> <StudentId> Ex: ERP PO Event VBCS 96 06
Identifier	Accept the default identifier value. The identifier is the same as the integration name you provided, but in upper case.
Version	Accept the default version number of 01.00.0000. Alternatively, if you want to change the version number, enter the version using numbers only in this format: xx.xx.xxxx.
What does this integration do?	This integration demonstrates the use of Oracle Integration ERP Cloud Adapter Eventing capability along with the REST adapter to store the PO record in VBCS PO Table
Which package does this integration belong to?	<i>Leave blank</i>

The screenshot shows the "Create Integration - Select a Style" dialog. The title bar says "Create Integration - Select a Style". Below it, a message says "How would you like to build your integration? Select a style to use." There are six options arranged in a grid:

- App Driven Orchestration**: Multi-step Integration flow triggered by an Application or API. This option is highlighted with a red box.
- Scheduled Orchestration**: Multi-step Integration flow triggered by a Schedule. Commonly used for Batch/Bulk Integrations or File processing.
- File Transfer**: Seamlessly and securely move files across the network.
- Basic Routing**: Basic App to App Routing with Data Mapping.
- Publish To OIC**: Publish messages from Apps to OIC Pub/Sub Channel.
- Subscribe To OIC**: Subscribe to messages from OIC Pub/Sub Channel.

Each option has a "Select" button below it. In the bottom right corner of the dialog, there is a "Cancel" button.

Create New Integration

?



Create New Integration

Enter information that describes this integration.

Describe this integration Use a meaningful name and description that will help others find and understand this integration. The Identifier and Version can be set only when the integration is created. The combination of Identifier and Version must be unique.

* What do you want to call your integration?

ERP PO Event VBCS 96 06

* Identifier

ERP_PO_EVENT_VBCS_96_06

* Version

01.00.0000

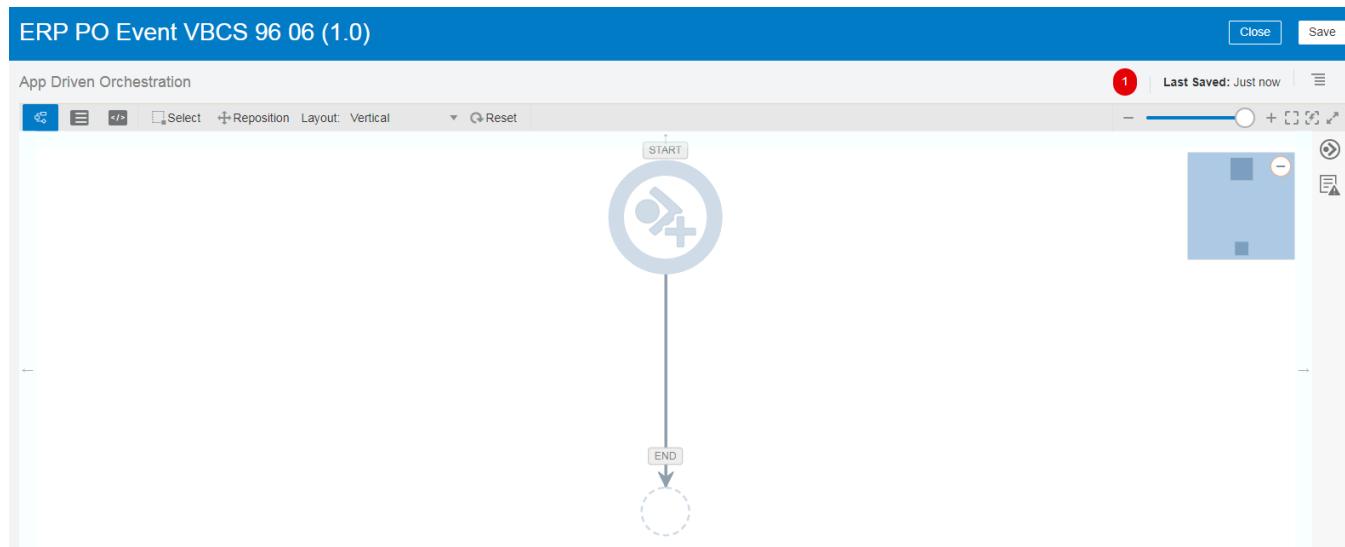
What does this integration do?

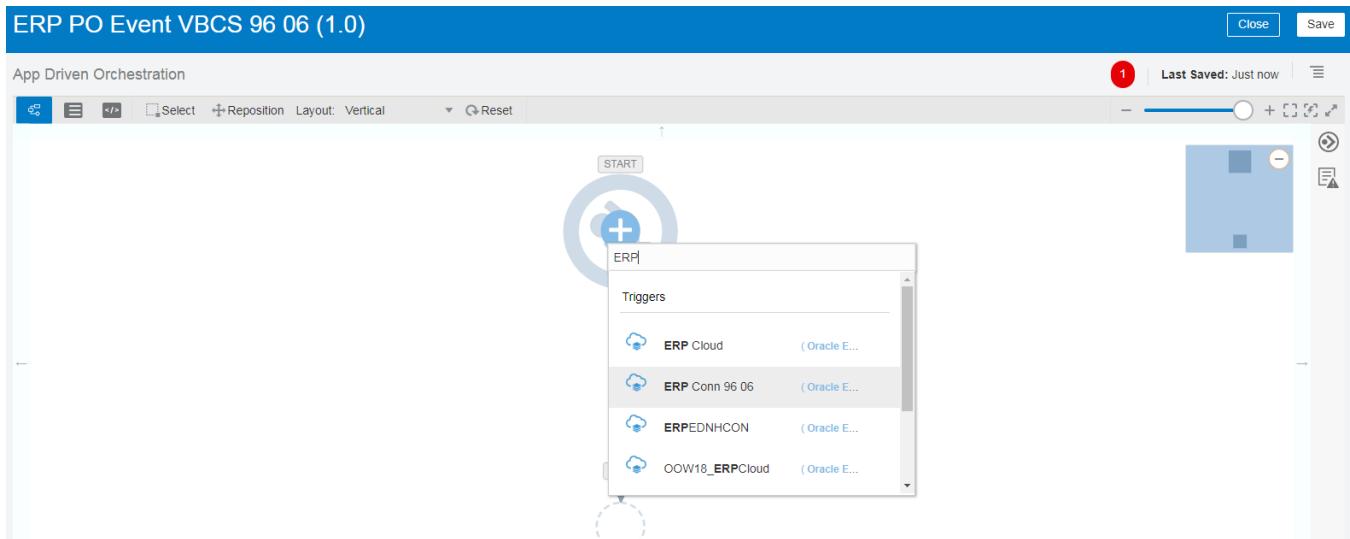
This integration demonstrates the use of Oracle Integration ERP Cloud Adapter Eventing capability along with the REST adapter to store the PO record in VBCS PO Table

Which package does this integration belong to?

Enter a new or existing package name

- Click **Create**. The integration canvas is displayed.





6. Hover over the circle next to Start, and click on the + sign
7. Select the ERP Connection (ERP conn 96 06) from the list which is pre created for you
8. Enter details as in the screenshot below to define the endpoint in the flow

Welcome to the Oracle ERP Cloud Endpoint Configuration Wizard
This wizard helps you configure an endpoint using the Oracle ERP Cloud connection. You will be asked to specify configuration parameters and define an operation for the service.

Basic Info

* What do you want to call your endpoint?
POEVENT

What does this endpoint do?
This will subscribe to the PO Events

9. In the next screen, select “With Business Events” as the option and search for the “Purchase Order Event” and provide an XPath Expression as below screenshot (Please copy it from XpathExpr.txt) and append Item description with your student number or initials) and Click Next

Configure Oracle ERP Cloud Endpoint

Help ▾ Back Next Cancel Done



Configure the Integration Service Endpoint to Receive Requests from the Oracle ERP Cloud Application

Select the business object or event that you want to receive from the Oracle ERP Cloud application as a request document to start this integration flow.

 Basic Info**Request**

Response

Summary

Configure a Request

 With Business Objects With Business Events Receive Status of ERP Import Job

* Business Event For Subscription

Purchase

Filter Expr for Purchase Order Event

Purchase Order Event

```
<xpathExpr
xmlns:ns0="http://xmlns.oracle.com/apps/prc/po/editDocument/purchaseOrderServiceV2"
xmlns:ns2="http://xmlns.oracle.com/apps/prc/po/editDocument/purchaseOrderServiceV2/types/">$eventPayload/ns2:result/ns0:Value/ns0:PurchaseOrderLine/ns0:itemDescription[lan Cable]</xpathExpr>
```

Event Description

The public business event is published when a purchase order is created, changed, finally closed, or reopened.

10. Select the Response type as “None” and Click Next

Configure Oracle ERP Cloud Endpoint

Help ▾ Back Next Cancel Done



Configure the Response Type to Send to the Oracle ERP Cloud Application

Select the business object to send as a response document from the integration flow to the Oracle ERP application. The immediate response option provides a request and response message exchange. The delayed response option provides an asynchronous response message exchange.

 Basic Info Request**Response**

Summary

Response Type

 Delayed None

Select the operation and business objects to configure for a successful callback response. You can also optionally select to configure a callback response for a failed integration flow.

You have selected to send no response document back to the source application. No further configuration is required.
Select Next to go to the Summary page.

11. Review summary page and Click Done

Configure Oracle ERP Cloud Endpoint

Help < Back Next > Cancel Done

Oracle Enterprise Resource Planning Business Event Subscription Summary
Oracle ERP Cloud endpoint configuration was successful.

POEVENT

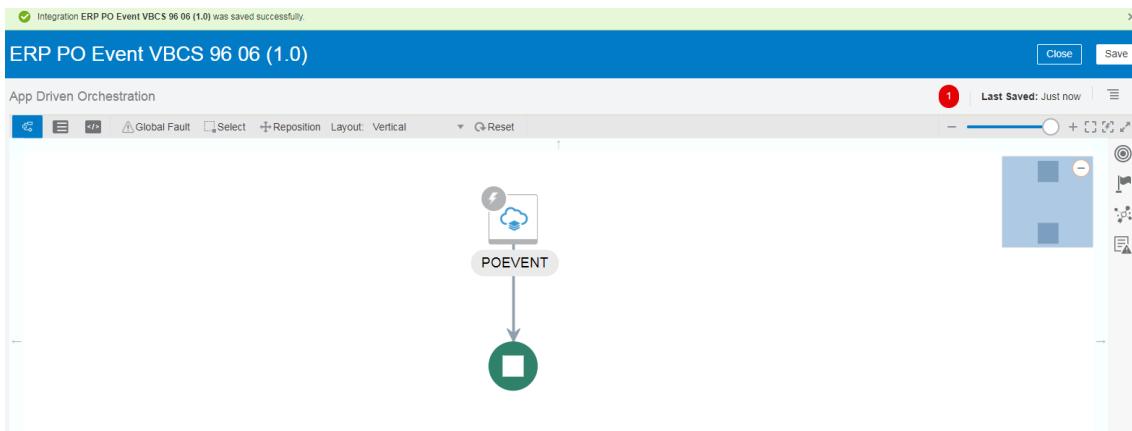
Description
This will subscribe to the PO Events

Subscribed Event: Purchase Order Event

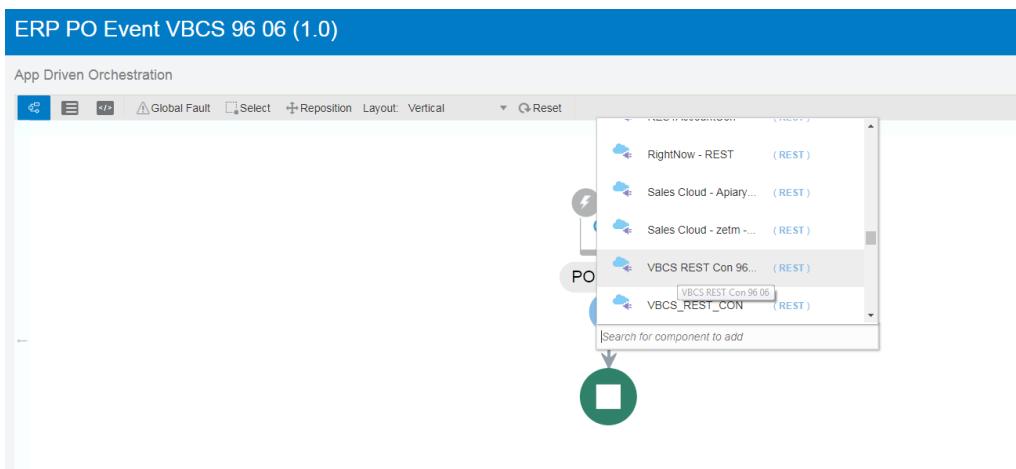
Filter Expression: <xpathExpr xmlns:ns0="http://xmlns.oracle.com/apps/prc/po/editDocument/purchaseOrderServiceV2/" xmlns:ns1="http://xmlns.oracle.com/apps/prc/po/editDocument/purchaseOrderServiceV2/">/ns0:purchaseOrder/ns0:header/ns0:orderNumber</xpathExpr>

Basic Info Request Response Summary

12. Integration so far



13. Next we will add the Invoke Activity Using the REST connection created earlier to verify if the PO exists in VBCS PO table. Hover on the wiring next to the “POEVENT” and click on “+” sign and Select the REST Connection (VBCS_REST_CON_<ClassId>_<StudentId>)



14. In the “Basic Info” page provide the details as per the screenshot below and Click Next

Configure REST Endpoint

Welcome to the REST Endpoint Configuration Wizard
This wizard helps you configure an endpoint using the REST adapter.

Basic Info

* What do you want to call your endpoint?
GETPO

What does this endpoint do?
Gets PO information from VBCS if exists

* What is the endpoint's relative resource URI?
/PO

* What action do you want to perform on the endpoint?
GET

Based on your selections, you can add parameters or configure a request and/or response for this endpoint.

Select any options that you want to configure:

Add and review parameters for this endpoint

Configure a request payload for this endpoint

Configure this endpoint to receive the response

Configure Request Headers? Standard Custom

15. In the Request Parameters page, Add a query parameter as per the below screenshot and Click Next. We wanted to search for a PO which can be invoked using below query parameter format

Ex:

https://<oid_host>/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO?q=pOHeaderId=300000074157561

Configure REST Endpoint

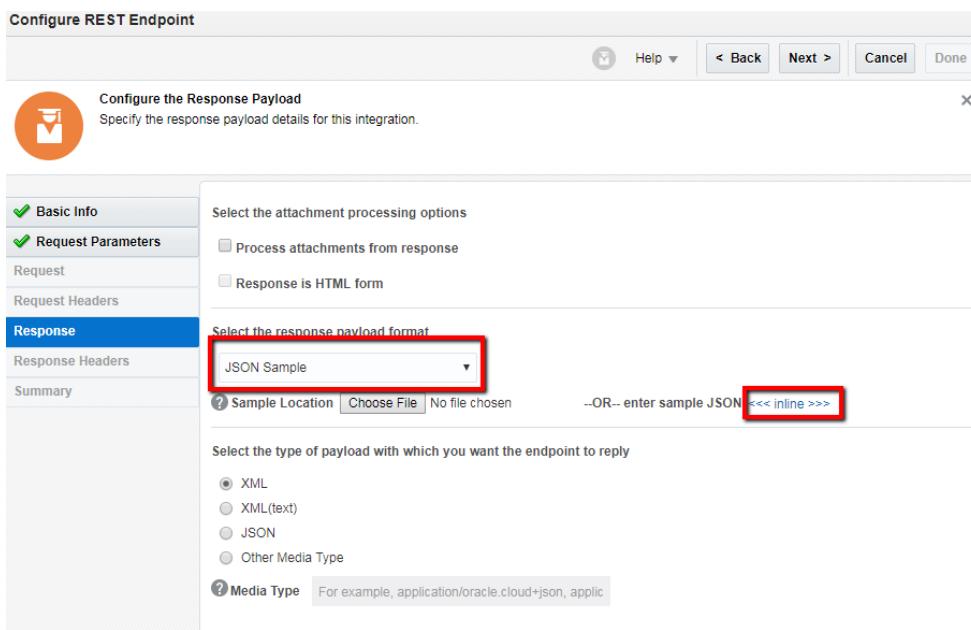
Configure the Request Query Parameters
Configure the request query parameters for this endpoint.

Request Parameters

Name	Data Type
q	string

Template Parameters
Displays the template parameters in the relative resource URI. Template parameters are determined by details you specified.

16. In the Response page Select the Response payload format as JSON



Select “inline” and enter the below JSON sample in the text area and Click OK. Select Next

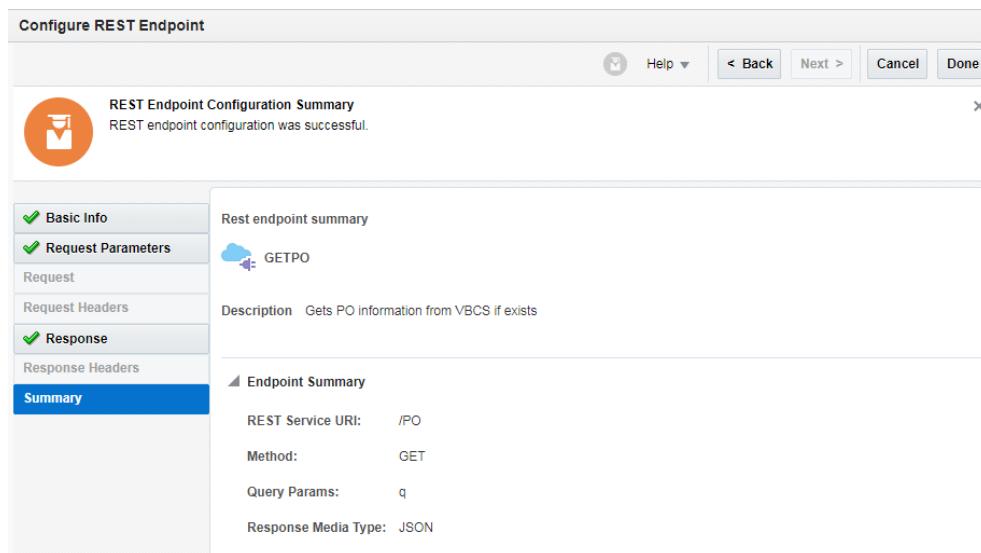
```
{
  "items": [
    {
      "id": 24,
      "creationDate": "2019-01-13T23:12:09+00:00",
      "lastUpdateDate": "2019-01-13T23:12:09.027+00:00",
      "createdBy": "john.doe@example.com",
      "lastUpdatedBy": "john.doe@example.com",
      "pOHeaderId": "300000074157561",
      "orderNumber": "162180",
      "procurementBUI": "300000046987012",
      "procurementBusinessUnit": "US1 Business Unit",
      "supplierId": "300000047414679",
      "supplier": "Dell Inc.",
      "soldToLegalEntity": "US1 Legal Entity",
      "IOCID": null,
      "links": [
        {
          "rel": "self",
          "href": "https://oic_host/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO/24",
          "name": "PO",
          "kind": "item",
          "properties": {
            "changeIndicator": "ACED0005737200136A6176612E7574696C2E41727261794C6973747881D21D99C7619D03000149000473697A65787
000000001770400000001737200106A6176612E6C616E672E446F75626C6580B3C24A296BFB0402000144000576616C
7565787200106A6176612E6C616E672E4E756D62657286AC951D0B94E08B02000078703FF00000000000000078"
          }
        },
        {
          "rel": "canonical",
          "href": "https://oic_host/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO/24",
        }
      ]
    }
  ]
}
```

```

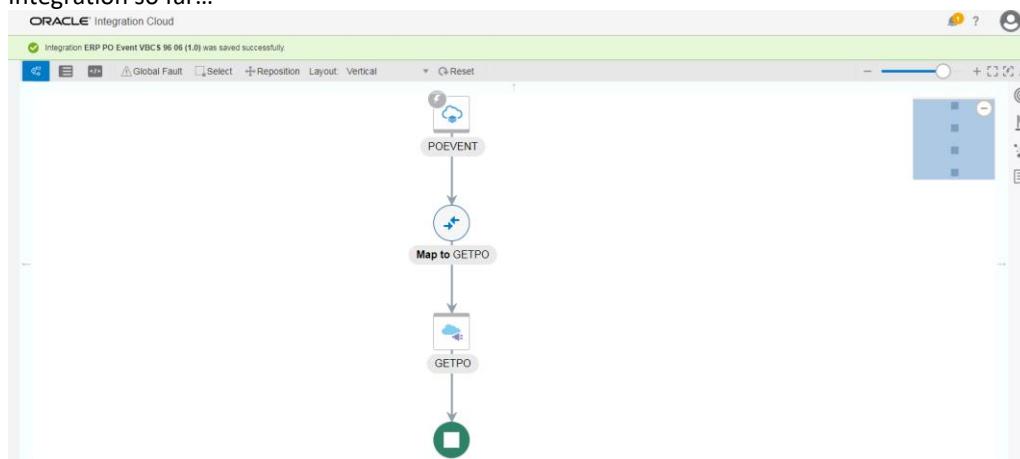
        "name": "PO",
        "kind": "item"
    }
]
}
],
"count": 1,
"hasMore": false,
"limit": 25,
"offset": 0,
"links": [
{
    "rel": "self",
    "href": "https://oic_host/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO",
    "name": "PO",
    "kind": "collection"
}
]
}
}

```

17. In the Summary page review the information and click Done

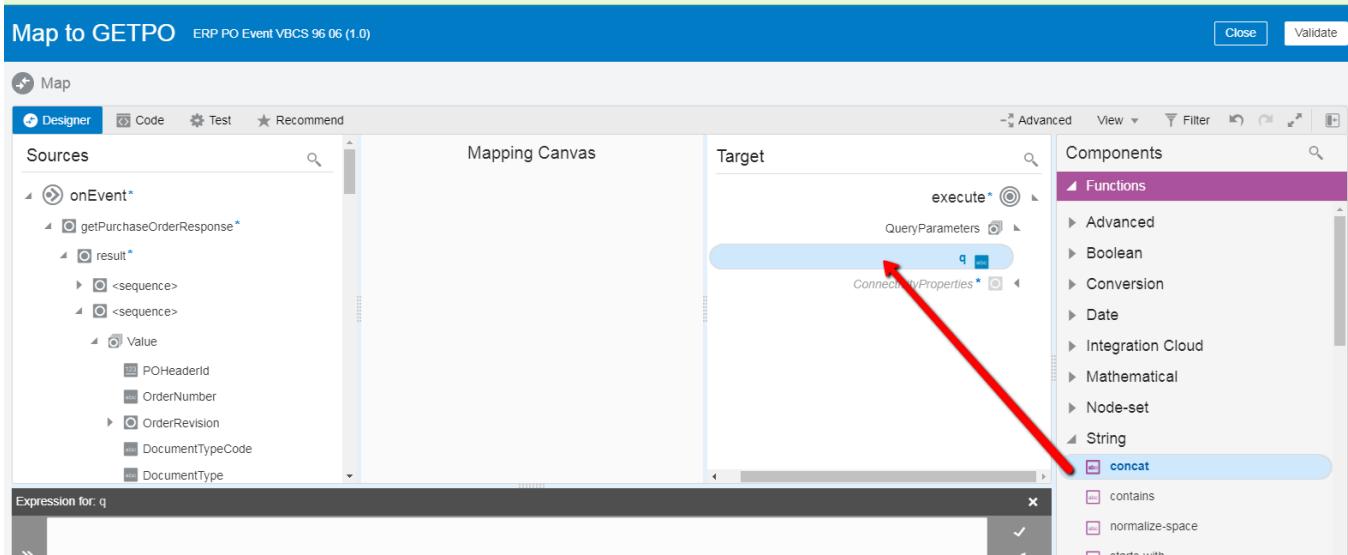


18. Integration so far...



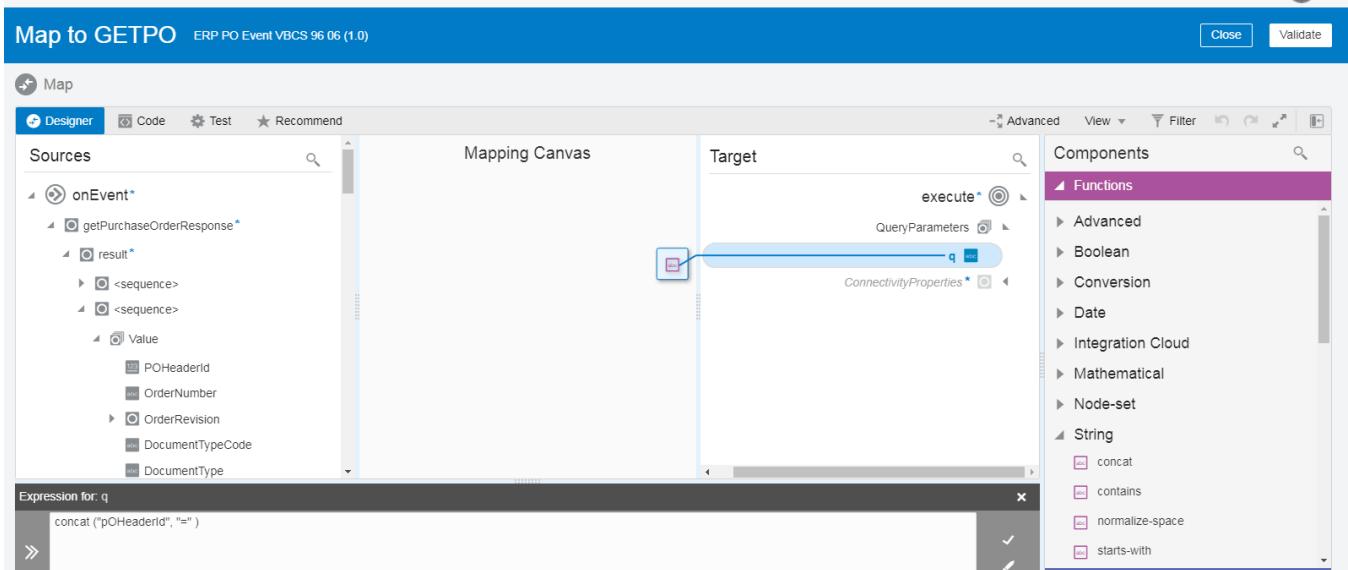
19. A Map activity (Map to GETPO) is created automatically. Perform the mapping as per below. Select the Map activity and Click on “Edit” to open the Mapper

Select the “q” Query Parameter from the Target, drag and drop the concat function to build query parameter value as “**pOHeaderId=300000074157561**”.



Provide static value “**pOHeaderId**” as the first parameter value for the concat(“”, “”, “”)

Provide static value “**=**” as the second parameter value for the concat(“”, “”, “”)



Provide the third parameter value by dragging the “POHeaderId” element inside the concat(“”, “”, “”)

Oracle Product Management

Map to GETPO ERP PO Event VBCS 96 06 (1.0)

Mapping Canvas

Sources

- onEvent*
- getPurchaseOrderResponse*
 - result*
 - <sequence>
 - <sequence>
 - Value
 - POHeaderId
 - OrderNumber
 - OrderRevision
 - DocumentTypeCode
 - DocumentType

Target

Components

Functions

- Advanced
- Boolean
- Conversion
- Date
- Integration Cloud
- Mathematical
- Node-set
- String
 - concat
 - contains
 - normalize-space
 - starts-with

execute*

QueryParameters

ConnectivityProperties*

Expression for: q
concat(poHeaderId, '=')

Click on the Tick mark in the expression window to save your expression

Map to GETPO ERP PO Event VBCS 96 06 (1.0)

Mapping Canvas

Sources

- onEvent*
- getPurchaseOrderResponse*
 - result*
 - <sequence>
 - <sequence>
 - Value
 - POHeaderId
 - OrderNumber
 - OrderRevision
 - DocumentTypeCode
 - DocumentType

Target

Components

Functions

- Advanced
- Boolean
- Conversion
- Date
- Integration Cloud
- Mathematical
- Node-set
- String
 - concat
 - contains
 - normalize-space
 - starts-with

execute*

QueryParameters

ConnectivityProperties*

Expression for: q
concat(poHeaderId, '=') /ssrcmpr:onEvent/inp1:getPurchaseOrderResponse/inp1:result/ns4:Value/ns4:POHeaderId

Click on Validate to check for any errors.

Map to GETPO ERP PO Event VBCS 96 06 (1.0)

Mapping is valid and ready to use.

Components

Functions

- Advanced
- Boolean
- Conversion
- Date
- Integration Cloud
- Mathematical

Mapping Canvas

Sources

- onEvent*
- getPurchaseOrderResponse*
 - result*
 - <sequence>
 - <sequence>
 - Value
 - POHeaderId

Target

Components

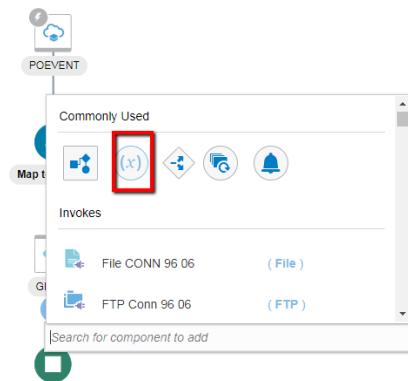
Functions

- Advanced
- Boolean
- Conversion
- Date
- Integration Cloud
- Mathematical

Save your integration

Integrating Your ERP with Oracle Integration

20. Now, we will assign PO Record Count value to a variable “POHeaderId”
 Hover on the wiring next to “GETPO” and click on “+” sign and Select Assign activity



Provide a name for the Assign Action and Click on Create

Create Action

Assign

Please give a unique name and description to this action.

* Name	PORecord
Description	Enter description

Create **Cancel**

Click on “+” to add a new Variable. Add a variable “PORecordCount” of Data type string

PORecord ERP PO Event VBCS 96 06 (1.0) **Close** **Validate**

(x) Assign

Assign variables to your integration. You can assign values to variables using the editor. Variable assignments can be a greater of complexity. For example, you can use assignments in other activities and in maps.

Add at least one named variable and specify its value by adding an expression.

Variable	Data Type	Description	Operation	Value
(x) PORecordCount	string	Type a description		Add an expression

+

Click the pencil icon to “Add an expression” to assign record count value. In the expression builder drag and drop the count element that we got as a response from the GETPO API call from the earlier step.

PORecordCount

Expression in "PORecord - ERP PO Event VBCS 96 06 (1.0)"

The screenshot shows the Oracle Integration Expression Builder interface. On the left, under 'Inputs', there is a tree view of source components: 'onEvent' > 'getPurchaseOrderResponse' > 'result'. Below it is 'SGETPO' > 'executeResponse' > 'response-wrapper' > 'items' > 'count'. A red arrow points from the 'count' node in the tree to the 'Expression' field on the right. The 'Expression' field contains the text '\$GETPO/nssrcmpr:executeResponse/nssrcdfi:response-wrapper/nssrcdfi:count'. At the bottom of the interface, there are 'View' and 'Components' tabs.

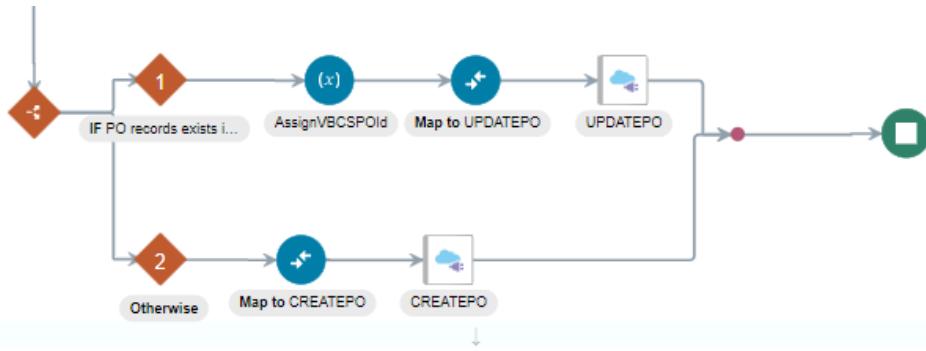
Validate and Close the Expression Builder...Save your integration

Integration so far...

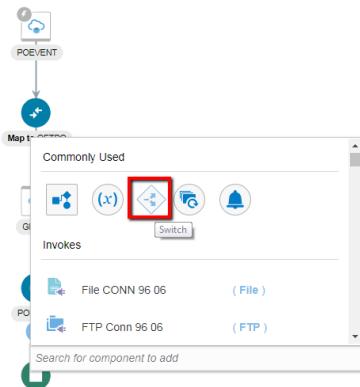


21. Next we will try to build an if-else logic as per below pseudo code

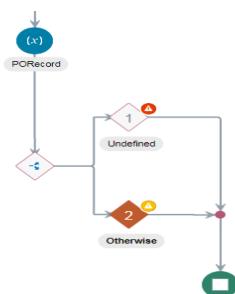
Search for PO record based on POHeaderID (Implemented already in previous steps)
 If (count of PORecords in VBCS) = 1
 - Update existing VBCS PO Record
 Else
 - Insert a new PO Record



Hover on the wiring next to the PORecord Activity and Select the “Switch” Activity



Below construct is created



If – Flow (Condition 1)

Select Condition “1” and click on Edit icon which brings up an Expression builder and provide the condition as per below

The screenshot shows the Expression builder interface with the following details:

- Title:** PO records exists in VBCS
- Expression Name:** PO records exists in VBCS
- Inputs:** A list of variables including \$PORRecordCount, \$Tracking_var_1, \$GETPO, executeResponse, response-wrapper, and result.
- Condition:** New Condition >> \$PORRecordCount = ''1''
- Buttons:** Close, Validate, Expression Mode, and a help icon.

22. Hover on the wiring next to the If Condition (1) activity and Click on “+” Sign and select the “Assign” Activity

Provide a name for the Assign Action as “AssignVBCSPOId”

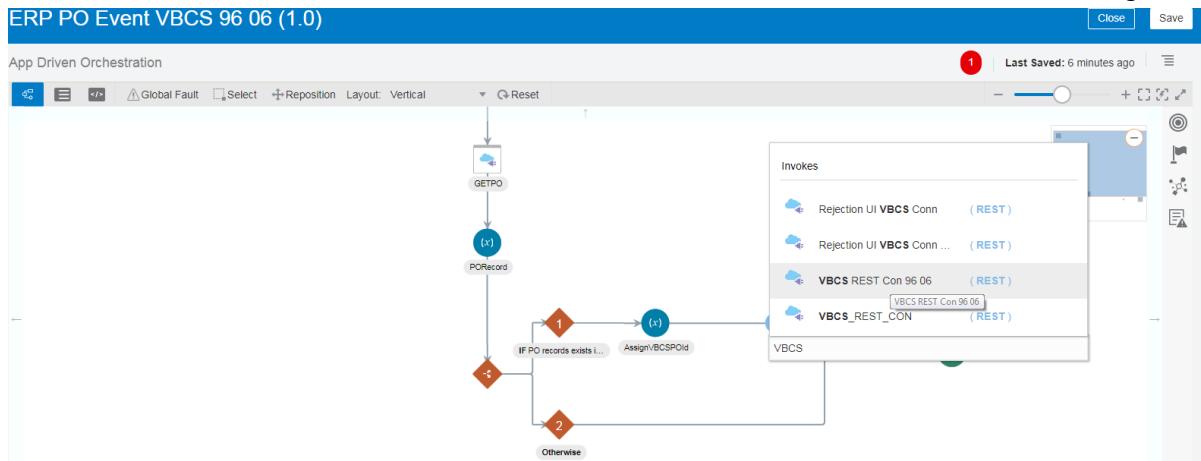
Create a Variable “VBCSPOId” of String data type and add an expression as below. Validate and Close the expression builder.

The screenshot shows the Expression Builder interface for the 'AssignVBCSPOId' action. The 'Expression' field contains the expression '\$GETPO/nssrcmp:executeResponse/nssrcdf:response-wrapper/nssrcdf:items/nssrcdf:id'. The 'Source' tree on the left shows the path: \$GETPO > executeResponse > response-wrapper > items > id. The 'id' node is selected and highlighted with a green checkmark. The 'Validate' button is visible at the top right.

The screenshot shows the 'AssignVBCSPOId' configuration window. It displays a table with one row for the variable 'VBCSPOId'. The table columns are: Variable, Data Type, Description, Operation, and Value. The 'Variable' column shows '(x) VBCSPOId', 'Data Type' shows 'string', 'Description' shows 'Type a description', 'Operation' shows 'id', and 'Value' shows 'id'. The 'Validate' button is visible at the top right.

Validate and Close the Assign window. Save your integration.

23. Hover on the wiring next to the Assign activity and Click on “+” Sign and select the REST Connection created earlier ex: VBCS_REST_CON_<ClassId>_<StudentId>



24. In the Basic Info Page provide the details as per below and Click Next

Configure REST Endpoint

Welcome to the REST Endpoint Configuration Wizard

This wizard helps you configure an endpoint using the REST adapter.

Basic Info

* What do you want to call your endpoint?
UPDATEPO

What does this endpoint do?
updates po with vbcS cross reference id

* What is the endpoint's relative resource URI?
/PO/{ID}

* What action do you want to perform on the endpoint?
PATCH ▾

Based on your selections, you can add parameters or configure a request and/or response for this endpoint.

Select any options that you want to configure:

- Add and review parameters for this endpoint
- Configure a request payload for this endpoint
- Configure this endpoint to receive the response

25. Nothing to Specify in Review Parameters page. Click Next

Configure REST Endpoint

Configure the Request Query Parameters
Configure the request query parameters for this endpoint.

Basic Info

Request Parameters (Selected)

Request Headers

Response

Response Headers

Summary

*** Resource URI /PO/{ID}**

Specify Query Parameters

[] Detach **[+]** **[X]**

Name	Data Type
No Configuration.	

Template Parameters

Displays the template parameters in the relative resource URI. Template parameters are determined by details you specified on the Basic Info page.

Name	ID	Type
		string

26. In the Request Page Select the request payload format as "JSON Sample". Select the "inline" link and provide the below Sample JSON request and Click Next

Configure REST Endpoint

Configure the Request Payload
Configure the request payload details for this endpoint.

Basic Info

Request Parameters

Request (Selected)

Request Headers

Response

Response Headers

Summary

Select the attachment processing options

Send attachments in request
 Request is HTML form

Select the request payload format

JSON Sample

Sample Location No file chosen
-OR- enter sample JSON <<< inline >>>

* Element request-wrapper

Select the type of payload with which you want the endpoint to send

XML
 XML(text)
 JSON
 URL-encoded
 Other Media Type

Media Type For example, application/oracle.cloud+json, applic

```
{
  "orderNumber": "163521",
  "procurementBuild": "300000046987012",
  "procurementBusinessUnit": "US1 Business Unit",
  "supplierId": "300000047414679",
  "supplier": "Dell.",
  "soldToLegalEntity": "US1 Legal Entity",
  "IOCID": 2
}
```

27. In the Response page Select the request payload format as "JSON Sample". Select the "inline" link and provide the below Sample JSON Response and Click Next

```
{
  "id": 24,
  "creationDate": "2019-01-13T23:12:09+00:00",
  "lastUpdateDate": "2019-01-22T16:46:08+00:00",
  "createdBy": "john.doe@example.com",
  "lastUpdatedBy": "john.doe@example.com",
  "pOHeaderId": "300000074157561",
  "orderNumber": "163521",
  "procurementBUIlD": "300000046987012",
  "procurementBusinessUnit": "US1 Business Unit",
  "supplierId": "300000047414679",
  "supplier": "Dell.",
  "soldToLegalEntity": "300000046973970",
  "IOClId": null,
  "links": [
    {
      "rel": "self",
      "href": "https://oic_host/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO/24",
      "name": "PO",
      "kind": "item",
      "properties": {
        "changeIndicator": "ACED0005737200136A6176612E7574696C2E41727261794C6973747881D21D99C7619D03000149000473697A65787
000000001770400000001737200106A6176612E6C616E672E446F75626C6580B3C24A296BFB0402000144000576616C
7565787200106A6176612E6C616E672E4E756D62657286AC951D0B94E08B020000787040000000000000000078"
      }
    },
    {
      "rel": "canonical",
      "href": "https://oic_host/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO/24",
      "name": "PO",
      "kind": "item"
    }
  ]
}
```

28. Review the Summary Page and Click on Done

Configure REST Endpoint

REST Endpoint Configuration Summary

REST endpoint configuration was successful.

Basic Info
Request Parameters
Request
Request Headers
Response
Response Headers
Summary

Description: updates po with vbcS cross reference id

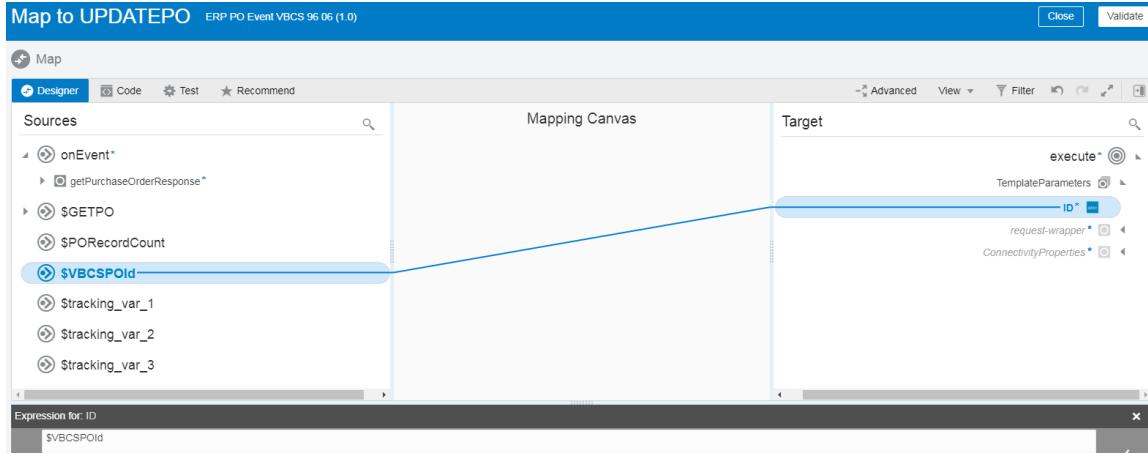
Endpoint Summary:

- REST Service URI: /PO/{ID}
- Method: PATCH
- Request Media Type: JSON
- Response Media Type: JSON

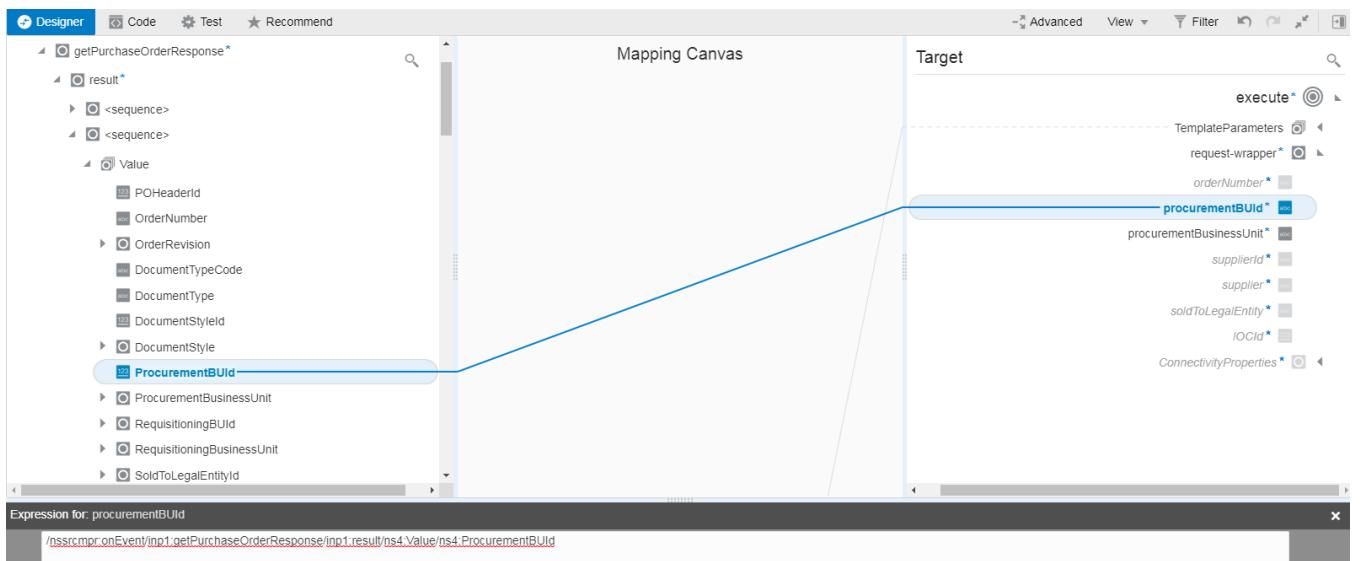
Oracle Product Management

29. A Map activity (Map to UPDATEPO) is created. Select the Map activity and Click on Edit Icon and provide mapping as per below screenshots

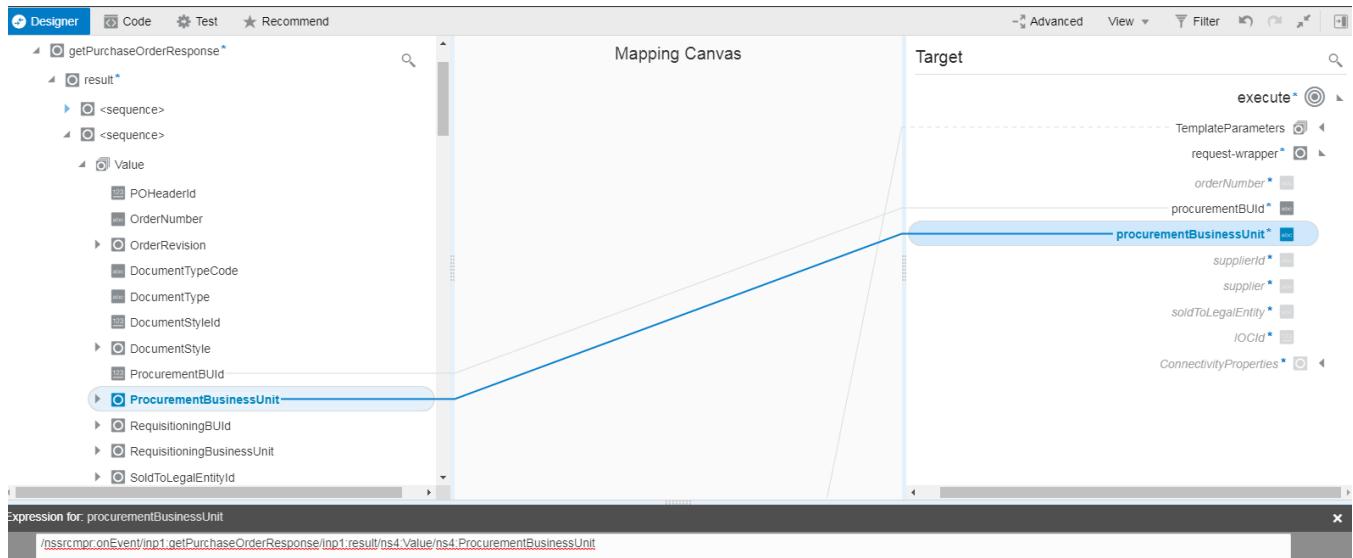
- a. Map \$VBCSPOId TO TemplateParameters-> ID



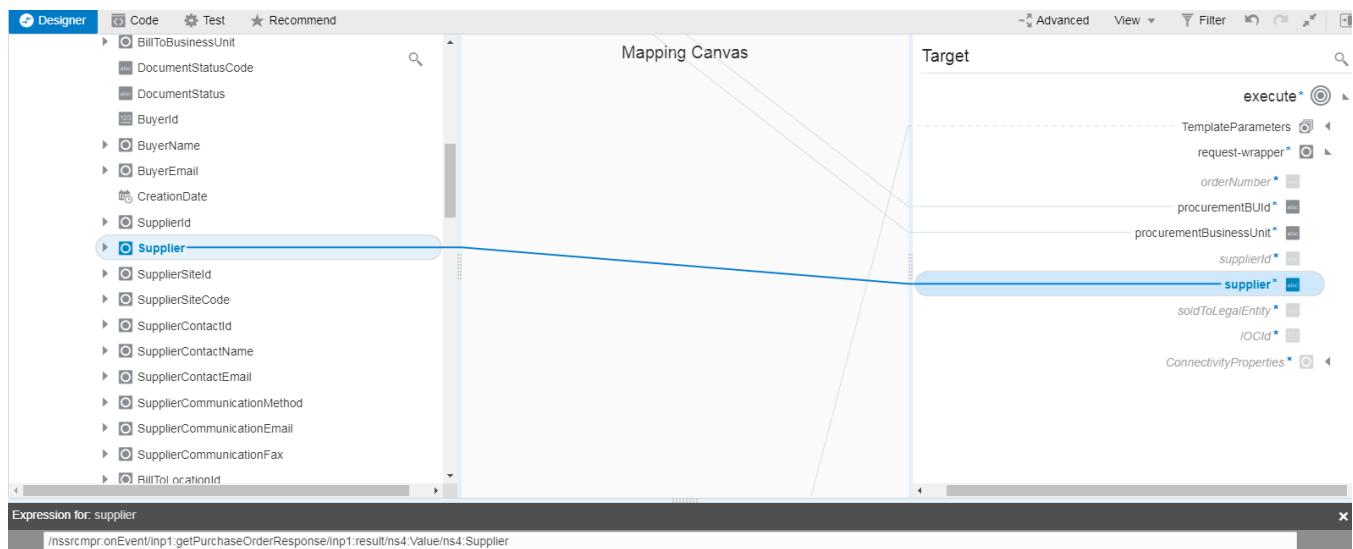
- b. Map onEvent -> getPurchaseOrderResponse->result-><sequence>->Value-> ProcurementBUID
TO request-wrapper -> procurementBUID



- c. Map onEvent -> getPurchaseOrderResponse->result-><sequence>->Value-> ProcurementBusinessUnit
TO request-wrapper -> procurementBusinessUnit

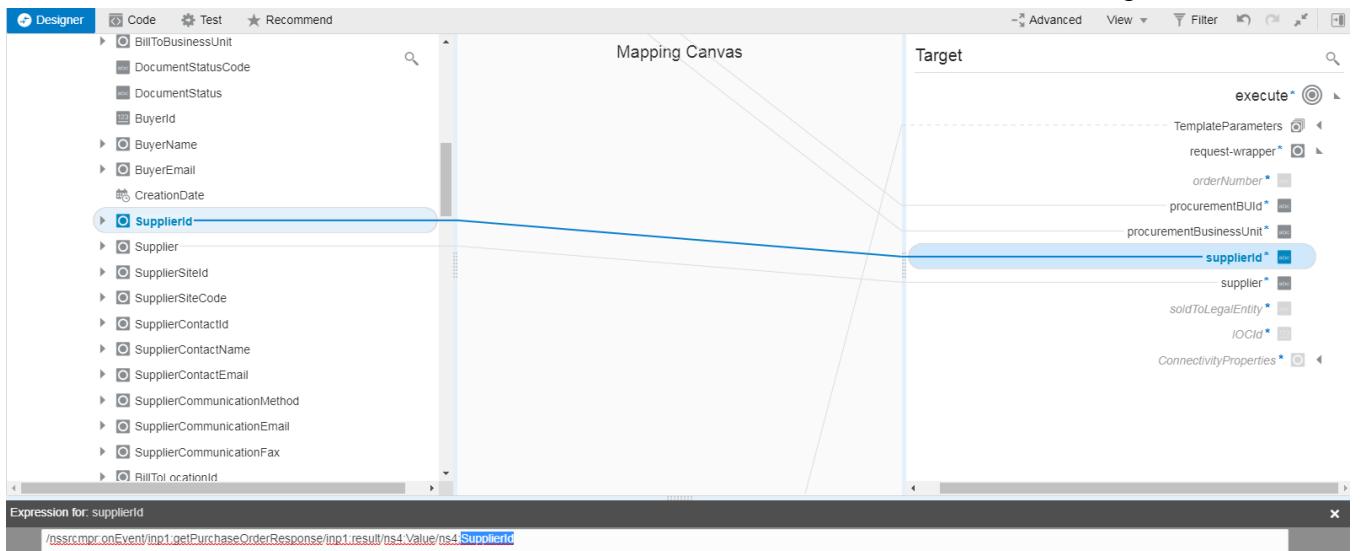


- d. Map onEvent -> getPurchaseOrderResponse->result-><sequence>->Value-> Supplier
TO request-wrapper -> supplier

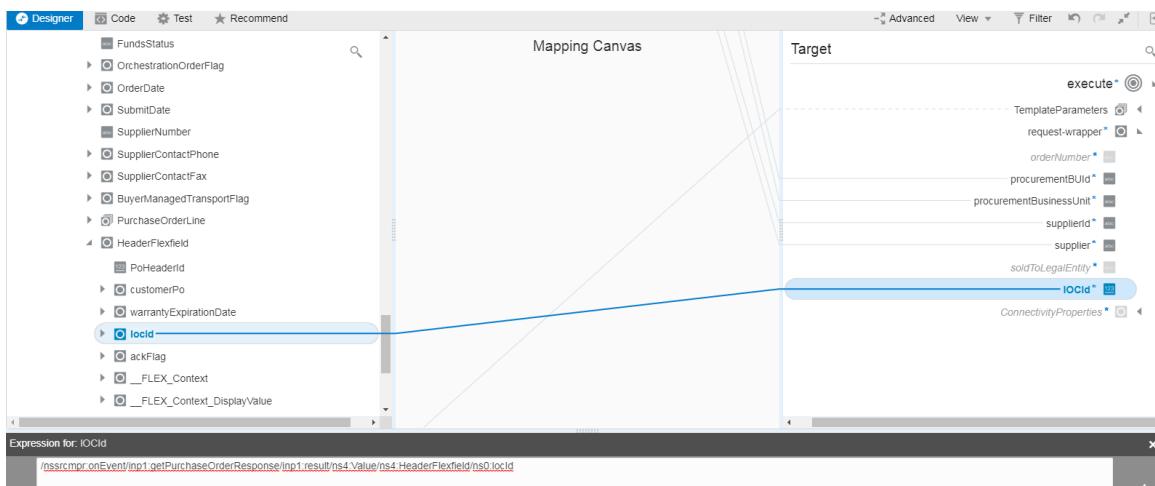


- e. Map onEvent -> getPurchaseOrderResponse->result-><sequence>->Value-> SupplierId
TO request-wrapper -> supplierId

Oracle Product Management

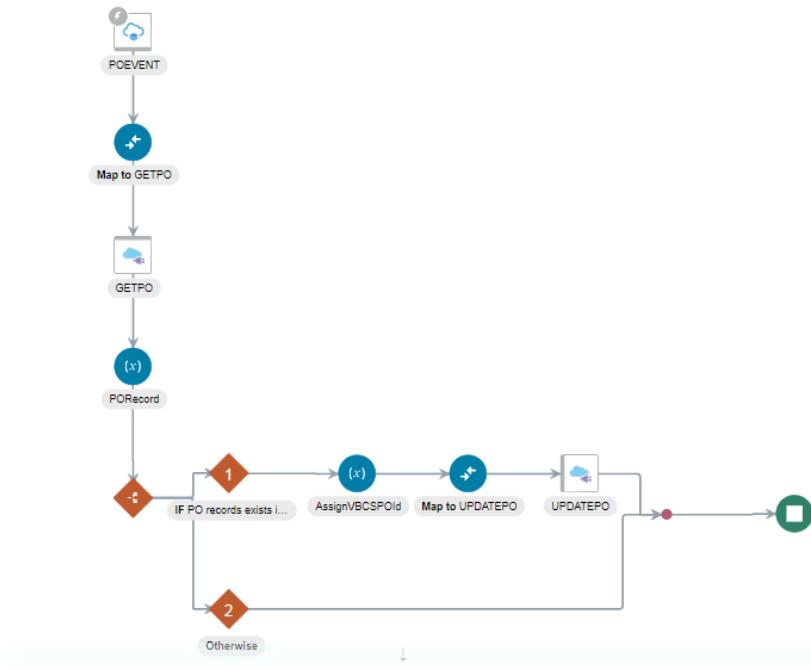


- f. Map onEvent -> getPurchaseOrderResponse->result-><sequence>->Value-> HeaderFlexfield->locId
TO request-wrapper -> IOId



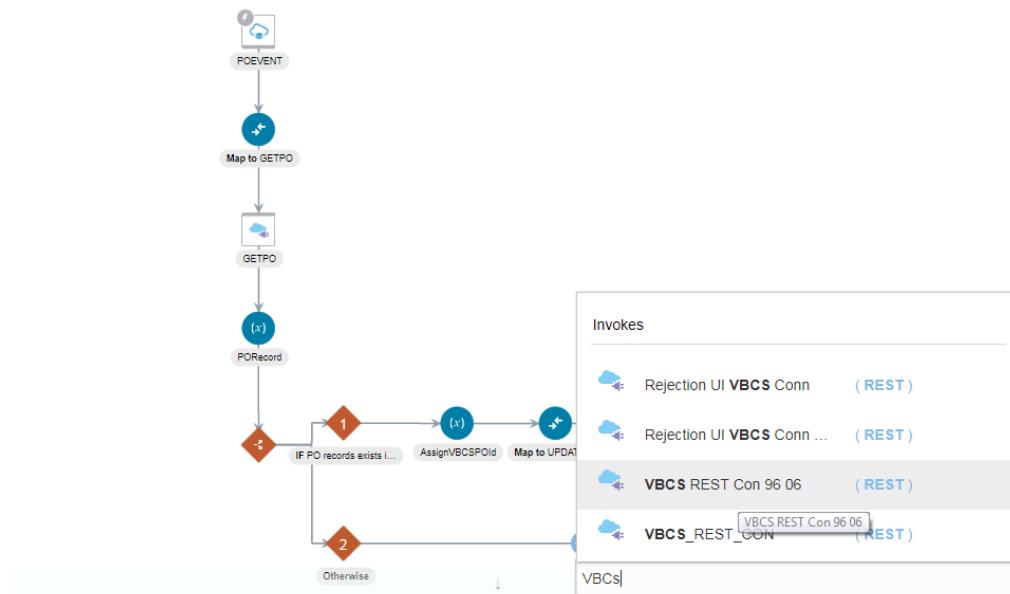
Click on Save and Validate. Close the mapper

30. Integration flow so far...

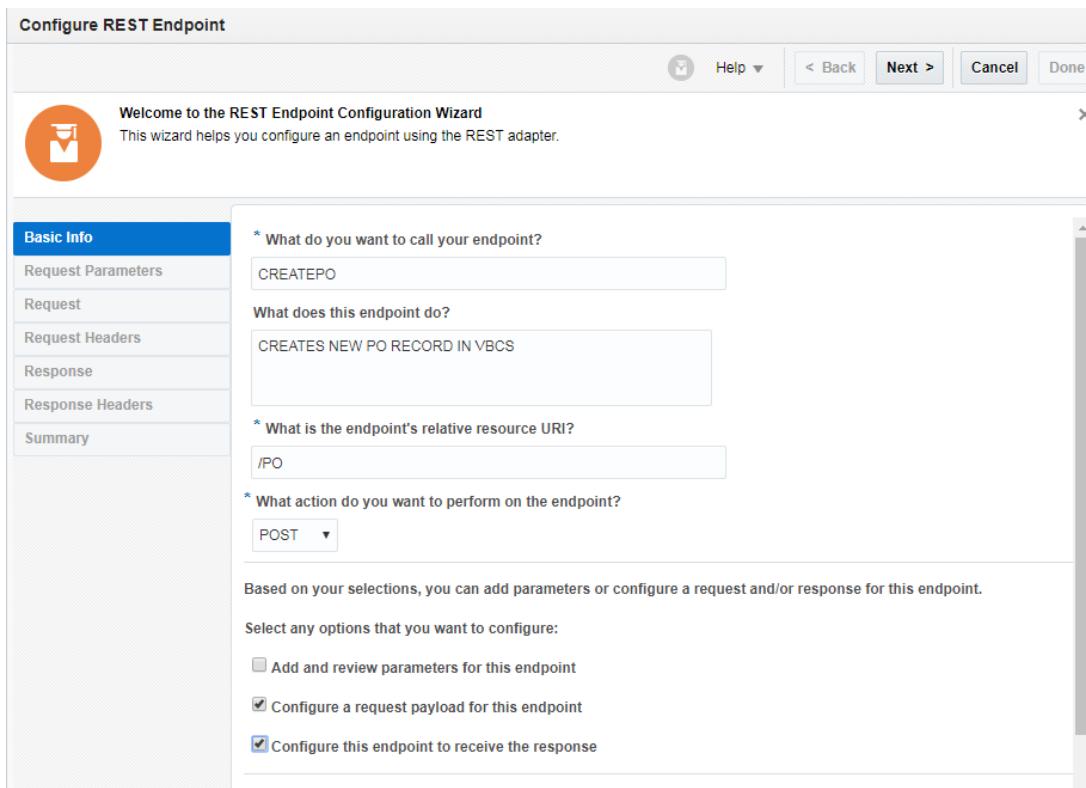


31. Else – Flow

Hover on the wiring next to “Otherwise” activity and Select VBCS_REST_CON_<ClassId>_<StudentId> connection



32. In Basic Page configure the details as per below



33. In the Request Page Select the request payload format as “JSON Sample”. Select the “inline” link and provide the below Sample JSON request and Click Next

```
{
  "pOHeaderId": "300000074157551",
  "orderNumber": "163521",
  "procurementBUId": "300000046987012",
  "procurementBusinessUnit": "US1 Business Unit",
  "supplierId": "300000047414679",
  "supplier": "Dell Inc.",
  "soldToLegalEntity": "US1 Legal Entity"
}
```

34. In the Response Page Select the response payload format as “JSON Sample”. Select the “inline” link and provide the below Sample JSON response and Click Next

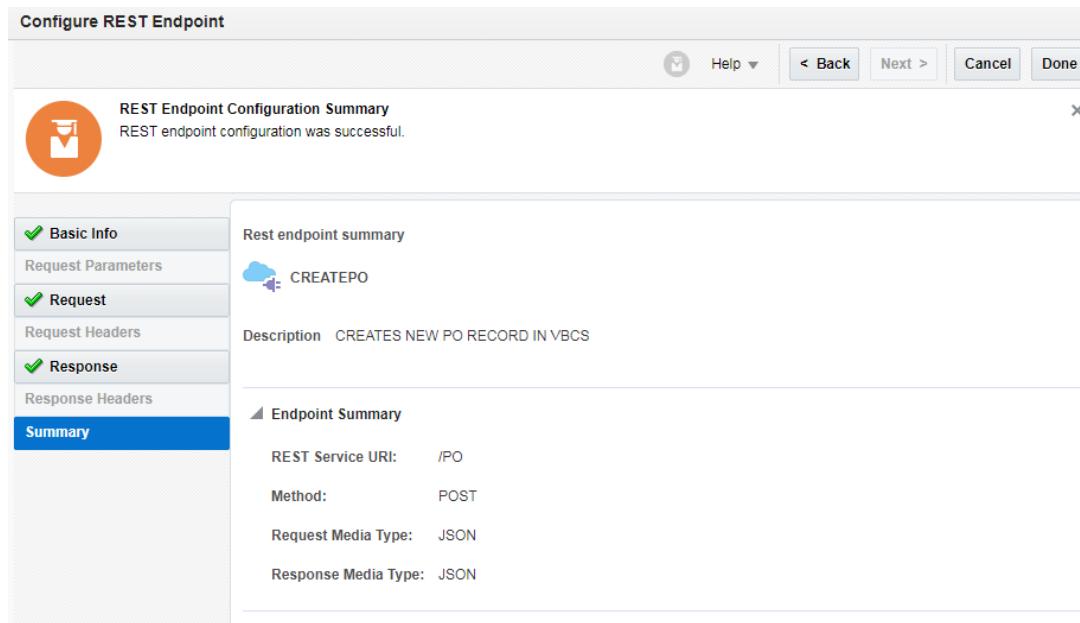
```
{
  "id": 41,
  "creationDate": "2019-01-22T17:32:24+00:00",
  "lastUpdateDate": "2019-01-22T17:32:24.027+00:00",
  "createdBy": "john.doe@example.com",
  "lastUpdatedBy": "john.doe@example.com",
  "pOHeaderId": "300000074157551",
  "orderNumber": "163521",
  "procurementBUId": "300000046987012",
  "procurementBusinessUnit": "US1 Business Unit",
  "supplierId": "300000047414679",
  "supplier": "Dell Inc.",
  "soldToLegalEntity": "300000046973970",
  "lOCLId": null,
```

```

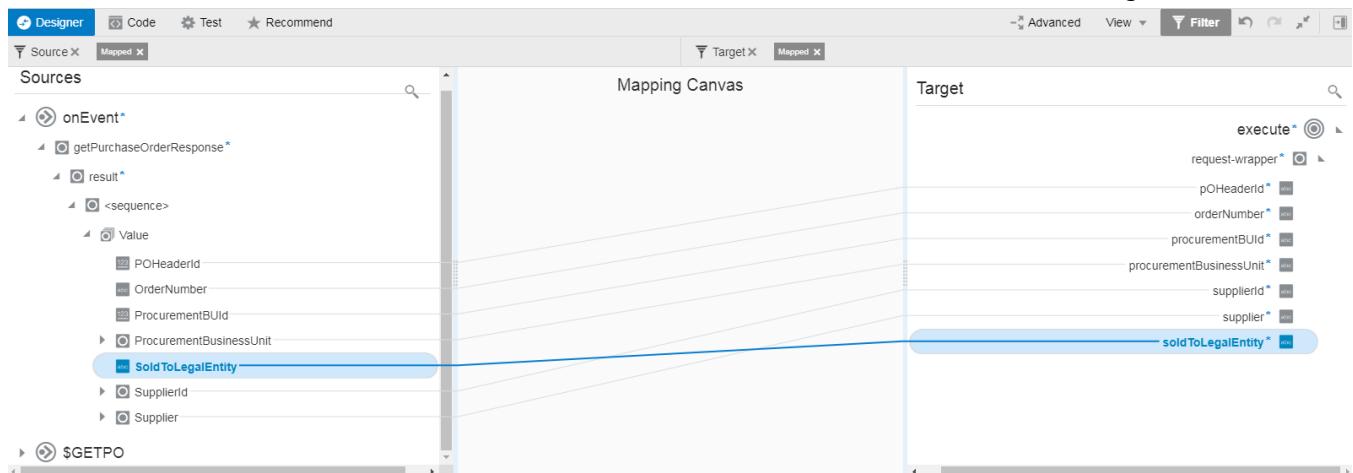
"links": [
  {
    "rel": "self",
    "href": "https://oichost/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO/41",
    "name": "PO",
    "kind": "item",
    "properties": {
      "changeIndicator":
        "ACED0005737200136A6176612E7574696C2E41727261794C6973747881D21D99C7619D03000149000473697A657870000
        00001770400000001737200106A6176612E6C616E672E446F75626C6580B3C24A296FB0402000144000576616C75657872
        00106A6176612E6C616E672E4E756D62657286AC951D0B94E08B02000078703FF00000000000000078"
    }
  },
  {
    "rel": "canonical",
    "href": "https://oichost/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO/41",
    "name": "PO",
    "kind": "item"
  }
]
}

```

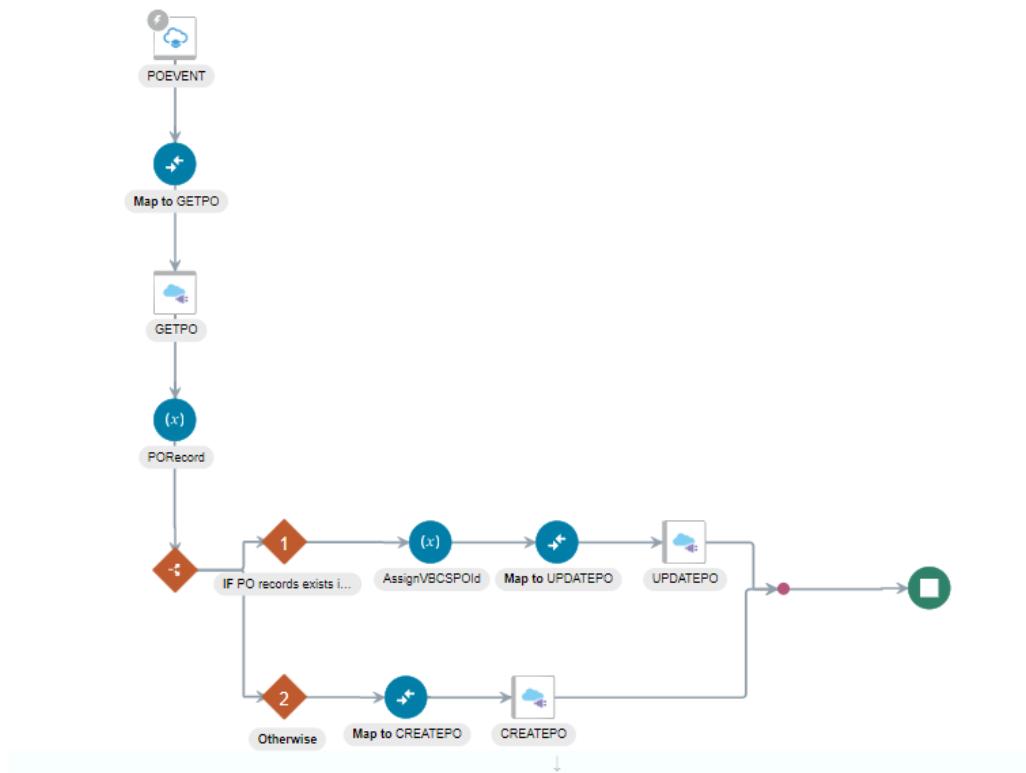
35. Review the Summary and Click Done



36. A Map activity is created. Provide the mapping for the Map Activity (Map to CREATEPO) as per the screenshot below



37. Final Flow 1 Integration



Save your integration flow.

38. While you are in the Integration flow page, Select the Hamburger icon on the TOP right corner and click on Tracking. Configure OrderNumber, POHeaderId and ItemDescription as tracking identifiers to identify your instance quickly when we a PO is created in ERP Cloud. Click Save on the Tracking page.

The screenshot shows the Oracle Product Management interface for configuring business identifiers. On the left, there's a sidebar with 'View', 'Filter', and 'Detach' buttons. Below that is a search bar labeled 'Source' with a 'Find...' button and a magnifying glass icon. A tree view shows the flow structure: 'onEvent' -> 'getPurchaseOrderResponse' -> 'result' -> 'Value' -> 'POHeaderId' (marked with a green checkmark) -> 'OrderNumber' (marked with a green checkmark). There are also sections for 'PurchaseOrderLine' and 'ItemDescription' (both marked with green checkmarks). To the right, a table lists tracking fields:

Primary	Tracking Field	Tracking Name	Tracking Variable	Help Text
<input checked="" type="checkbox"/>	OrderNumber	Order Number	tracking_var_1	How to track it?
	POHeaderId	PO Header Id	tracking_var_2	How to track it?
	ItemDescription	Item Description	tracking_var_3	How to track it?

Activate Integration (Flow1)

1. Click on 'Activate' option on the integration "ERP PO EVENT VBCS <ClassId> <StudentId>"

The screenshot shows the Oracle Integration interface for managing integration flows. At the top, there are icons for cloud-to-cloud integration. The title bar reads 'ERP PO Event VBCS 96 06 [NEW]'. In the center, it says '1.0.0' and 'Sep 4, 2019 08:13:35 PM IST'. On the right, there are several buttons: a pencil icon, a power icon, a list icon, and a dropdown arrow. The 'Activate' button is highlighted with a red box.

2. Check the *Oracle Recommends* check box to contribute your mappings to the Recommendations engine that then suggests mappings to you and other users for similar integrations. Also, check *Enable tracing* and *Enable payload* for debugging and troubleshooting (you would typically have the tracking and payload options disabled on production instances, but we enable them here for this lab).

Activate Integration



ERP PO Event VBCS 96 06 (1.0)

Oracle Recommends

Contribute integration mappings to Oracle Recommends.



Oracle Integration leverages the collective intelligence to recommend which fields should be mapped while developing an integration. These recommendations are built based on the mappings contributed to Oracle Recommends anonymously. Unselect the checkbox if you do not wish to contribute the mappings. You may change this in recommendations page from settings menu.

Oracle Asserter: When Asserter recording is enabled, payloads will be captured and integration instances will be recorded. Recordings can be played later and maximum five recordings will be maintained for an integration.

Enable Asserter Recording

Tracing: When tracing is enabled, integration activity can be viewed in the Activity Stream.

Enable Tracing

Include Payload



When payload is included, sensitive information from the payload is written into log files, which can be downloaded and viewed. This may pose a security risk, and also slow down your system. Not recommended in a production environment.

Cancel

Activate

- The activation should complete in a few seconds typically.



ERP PO Event VBCS 96 06 NEW

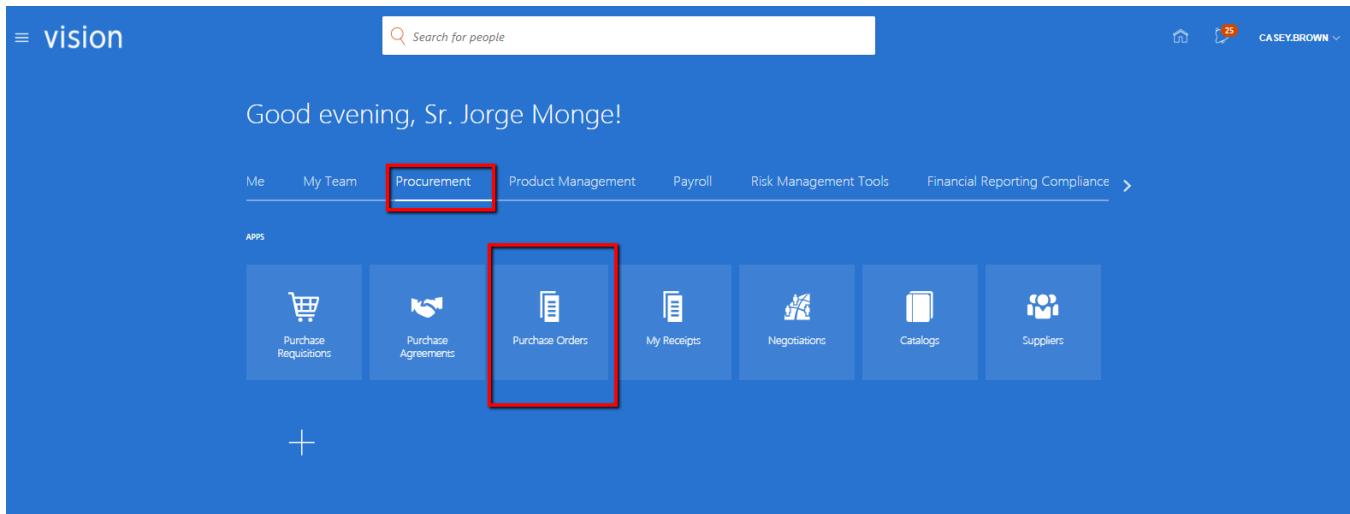
1.0.0

Sep 4, 2019 08:17:06 PM IST

Active

Testing the ERP Event Flow Integration (Flow1)

- Login into ERP Cloud as casey.brown with the credentials provided
- Select Procurement tab and click on Purchase Orders



3. Select Tasks tab and Click on Create Order

The screenshot shows the Oracle Product Management Overview page. On the left, there are several cards: "Orders Requiring Attention" (0 Failed Submission, 1 With invoice Holds, 0 Rejected), "Orders in Process" (No data to display), "Recent Activity" (listing recent changes like "Change order submitted | 1/13/19 Order 163521 - Dell Inc."), "Requisition Lines" (0 Requiring Action), and "Incomplete Orders" (0 Orders, 2 Change C...). On the right, a sidebar titled "Tasks" contains a list of actions grouped by category: "Requisitions" (Process Requisitions, Import Requisitions, View Requisition Lines), "Orders" (Manage Orders, Create Order, Generate Orders, Import Orders), "Agreements" (Manage Agreements, Create Agreement, Import Blanket Agreements, Import Contract Agreements, Initiate Retroactive Price Update, Generate Approved Supplier List Entries), "Deliverables" (Manage Deliverables), "Supply Base" (Research Suppliers, Manage Suppliers, Manage Approved Supplier List Entries), and "Administration" (Communicate Purchasing Documents, Reassign Purchasing Documents, Close Schedules in Spreadsheet, Purge Processed Open Interface Documents, Manage Procurement Agents, Manage Buyer Assignment Rules). The "Create Order" button under the "Orders" section is highlighted with a red box.

4. Enter Supplier information (Dell Inc.) and leave the rest to default. Click on Create

Create Order

* Style	Purchase Order
* Procurement BU	US1 Business Unit
* Requisitioning BU	US1 Business Unit
Supplier	Dell Inc.
Supplier Site	Dell US1
Supplier Contact	Gill, Noah
Default Ship-to Location	Seattle
Sold-to Legal Entity	US1 Legal Entity
* Currency	USD
* Buyer	Brown, Casey
<input type="button" value="Create"/> <input type="button" value="Cancel"/>	

5. In the Purchase Order page go to the Order Lines Section and provide Item information as below

Edit Document (Purchase Order): 163531

Order	163531	Communication Method	None	Total	5.00 USD																										
Status	Incomplete	Bill-to Location	Seattle	Procurement Card																											
* Buyer	Brown, Casey	Default Ship-to Location	Seattle	Description																											
Creation Date	1/25/19			Requisition																											
				Agreement	S2167																										
Terms Notes and Attachments																															
Required Acknowledgment: None Acknowledge Within Days: <input type="text"/> Payment Terms: Net 30																															
Shipping Method: UPS Freight Terms: Buyer pays freight FOB: Origin																															
<input type="checkbox"/> Requires signature <input type="checkbox"/> Buyer managed transportation <input checked="" type="checkbox"/> Pay on receipt <input type="checkbox"/> Confirming order																															
Additional Information																															
Customer PO: <input type="text"/> Warranty Expiration Date: <input type="text"/> mm/dd/yy																															
Letter of credit Id: <input type="text"/> Supplier Acknowledgment Flag: <input type="checkbox"/>																															
Lines Schedules Distributions																															
<table border="1"> <thead> <tr> <th>Actions</th> <th>Line</th> <th>Type</th> <th>Edit</th> <th>Item</th> <th>Description</th> <th>Category Name</th> <th>Quantity</th> <th>UOM</th> <th>Price</th> <th>On</th> <th>Location</th> <th>Requester</th> </tr> </thead> <tbody> <tr> <td></td> <td>1</td> <td>Goods</td> <td><input type="button" value="Edit"/></td> <td><input type="button" value="Item"/></td> <td>Lan Cable</td> <td>Computer Supplies</td> <td>1</td> <td>Ea</td> <td>\$0.00</td> <td>...</td> <td>Seattle</td> <td><input type="text"/></td> </tr> </tbody> </table>						Actions	Line	Type	Edit	Item	Description	Category Name	Quantity	UOM	Price	On	Location	Requester		1	Goods	<input type="button" value="Edit"/>	<input type="button" value="Item"/>	Lan Cable	Computer Supplies	1	Ea	\$0.00	...	Seattle	<input type="text"/>
Actions	Line	Type	Edit	Item	Description	Category Name	Quantity	UOM	Price	On	Location	Requester																			
	1	Goods	<input type="button" value="Edit"/>	<input type="button" value="Item"/>	Lan Cable	Computer Supplies	1	Ea	\$0.00	...	Seattle	<input type="text"/>																			
Columns Hidden: 45																															

Note: For "Description" enter a Unique Value of your choice, which was provided in the xpath expression earlier when configuring the ERP adapter PO Event

Ref:

Configure Oracle ERP Cloud Endpoint

Configure the Integration Service Endpoint to Receive Requests from the Oracle ERP Cloud Application
Select the business object or event that you want to receive from the Oracle ERP Cloud application as a request document to start this integration flow.

Basic Info

Request

Response

Summary

Configure a Request

With Business Objects With Business Events Receive Status of ERP Import Job

* Business Event For Subscription: Purchase

Filter Expr for Purchase Order Event: Purchase Order Event

```
<xpathExpr
xmlns:ns0="http://xmlns.oracle.com/apps/prc/po/editDocument/purchaseOrderServiceV2/"
xmlns:ns2="http://xmlns.oracle.com/apps/prc/po/editDocument/purchaseOrderServiceV2/types/">$eventPayload/ns2:r
esult/ns0/Value/ns0:PurchaseOrderLine/ns0:ItemDescription
ns0:Lan Cable</xpathExpr>
```

Event Description

The public business event is published when a purchase order is created, changed, finally closed, or reopened.

6. Click on Submit to Create a Purchase Order. Make a note of the Order Number

Confirmation

The document (Purchase Order) 163531 was submitted for approval.

OK

7. Click on Tasks tab and Select Manage Order

vision

Overview

Orders Requiring Attention: 0 Failed Submission, 1 With Invoice Holds, 0 Rejected

Orders In Process: No data to display

Recent Activity: Change order submitted | 1/13/19 Order 163531 - Dell Inc.

Requisition Lines: 0 Requiring Action

Incomplete Orders: 0 Orders

Aging Requisition Lines: Days Unprocessed: 0 to 3 Days, 4 to 7 Days, 8 to 14 Days, Over 14 Days

Open Schedules: 0 Open, 0 Closed for Receiving, 0 Closed for Invoicing, 0 On Hold

Navigation menu:

- Requisitions
 - Process Requisitions
 - Import Requisitions
 - View Requisition Lines
- Orders
 - Manage Order** (highlighted)
 - Create Order
 - Generate Orders
 - Import Orders
- Agreements
 - Manage Agreements
 - Create Agreement
 - Import Blanket Agreements
 - Import Contract Agreements
 - Initiate Retrospective Price Update
 - Generate Approved Supplier List Entries
- Deliverables
 - Manage Deliverables
- Supply Base
 - Research Suppliers
 - Manage Suppliers
 - Manage Approved Supplier List Entries
- Administration
 - Communicate Purchasing Documents
 - Resign Purchasing Documents
 - Close Schedules in Spreadsheet
 - Purge Processed Open Interface Documents
 - Manage Procurement Agents
 - Manage Buyer Assignment Rules

8. Search for the Order Number and wait for the Status field to change to “Open”

The screenshot shows the Oracle Product Management interface. In the search bar, the order number '163531' is entered. The search results table has columns: Order, Description, Supplier, Ordered, Currency, Status, Funds Status, Change Order Funds Status, Life Cycle, Change Order, Creation Date, Supplier Site, Supplier Contact, Acknowledgment Due Date, and Shipping Method. The first result for order 163531 shows Dell Inc. as the supplier, 5.00 USD as the amount, and the status as 'Open'.

9. Go back to Oracle Integration -> Monitoring -> Integrations → Tracking Page and Search for your instance with the Order Number as Tracking identifier

The screenshot shows the Oracle Integration Monitoring interface. On the left sidebar, 'Tracking' is selected. The main area shows a tracking instance for Order Number 163531, Instance ID 10800099, and PO Header Id 300000185555682. The status is 'COMPLETED'. The instance was received 2 minutes ago and completed 2 minutes ago with a duration of 12 seconds.

10. Click on the instance and verify that the Flow is completed successfully

The screenshot shows a Visual Builder flow for a PO record. The flow starts with a 'POSTVENT' step, followed by a 'Map to GETPO' step. This leads to a 'GETPO' step, which then connects to a decision diamond 'IF PO record exists'. If the condition is true, the flow goes through 'Assign/CREATEPO' and 'Map to UPDTEPO' steps, followed by an 'UPDATEPO' step. If the condition is false ('Otherwise'), it goes through 'Map to CREATEPO' and 'CREATEPO' steps. The flow ends with a green terminal node. A note indicates the flow is completed 43 minutes ago.

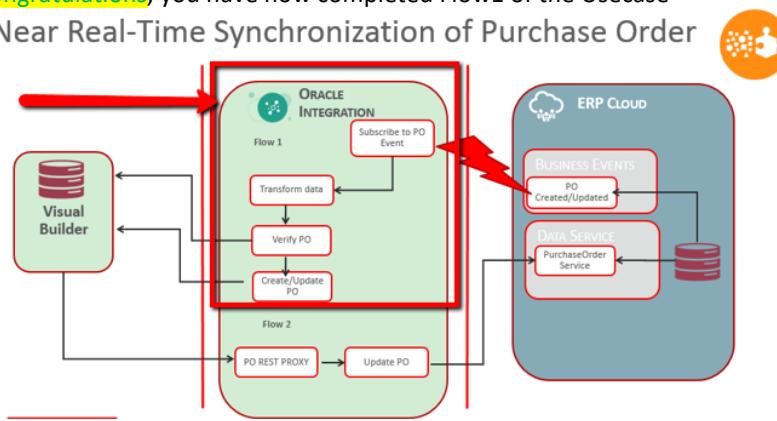
Since it is a new PO record which does not exist in VBCS “Otherwise” flow path is chosen based on the condition

11. Navigate to Visual Builder and Click on imported application
12. Select the Business Objects Tab and click on “PO” object you should see a new record is created

The screenshot shows the Oracle Visual Builder interface with a table titled 'PO' containing one row of data. The columns are: Id (id), LOCId (LOCId), OrderNumber (orderNumber), POHeaderId (pOHeaderId), ProcurementBUId (procurementBUId), and ProcurementBusinessUnit (procurementBusinessUnit). The data in the table is:

Id (id)	LOCId (LOCId)	OrderNumber (orderNumber)	POHeaderId (pOHeaderId)	ProcurementBUId (procurementBUId)	ProcurementBusinessUnit (procurementBusinessUnit)
63		163531	300000176127931	30000046987012	US1 Business Unit

Congratulations, you have now completed Flow1 of the Usecase
Near Real-Time Synchronization of Purchase Order



You leveraged rich capabilities of Oracle Integration such as ERP Adapter Event Subscription Capabilities using and App-driven (trigger-based) integration, ERP Cloud adapter, Rest adapter, Data Mapper, Actions such as Invoke and Activities such as Assign, Map, If-Otherwise etc, Configuring Business Tracking Identifiers and monitoring running flows.

You could now leverage this knowledge to design, activate and monitor several use cases for ERP Real time Synchronization

Creating Connections (Flow2)

The following Connections have been created and configured. You will be using these connections for creating Integration flow

Connection Name	Connection Type
PO REST Interface Con 96 06	Rest Adapter
ERP Conn 96 06	ERP Adapter

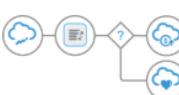
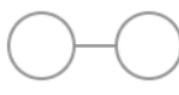
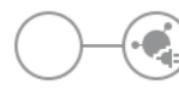
Creating the PO Proxy Integration (Flow2)

1. On the Oracle Integration home page, click **Integrations**.
2. On the Integrations page, click **Create**. The Create Integration - Select a Style/Pattern dialog is displayed.
3. Select **App Driven Orchestration** type of integration. The **Create New** Integration dialog is displayed.
4. Enter the following information:

Field Element	Value
What do you want to call your integration?	ERP PO Proxy <ClassId> <StudentId> Ex: ERP PO Proxy 96 06
Identifier	Accept the default identifier value. The identifier is the same as the integration name you provided, but in upper case.
Version	Accept the default version number of 01.00.0000. Or, if you want to change the version number, enter the version using numbers only in this format: xx.xx.xxxx.
What does this integration do?	This integration demonstrates the use of Oracle Integration REST adapter and ERP Cloud Adapter to create a REST to SOAP proxy for ERP Cloud webservices
Which package does this integration belong to?	Leave blank

Create Integration - Select a Style

How would you like to build your integration? Select a style to use.

App Driven Orchestration  Multi-step Integration flow triggered by an Application or API. Select	Scheduled Orchestration  Multi-step Integration flow triggered by a Schedule. Commonly used for Batch/Bulk Integrations or File processing. Select	File Transfer  Seamlessly and securely move files across the network. Select
Basic Routing  Basic App to App Routing with Data Mapping. Select	Publish To OIC  Publish messages from Apps to OIC Pub/Sub Channel. Select	Subscribe To OIC  Subscribe to messages from OIC Pub/Sub Channel. Select

Cancel

Create New Integration

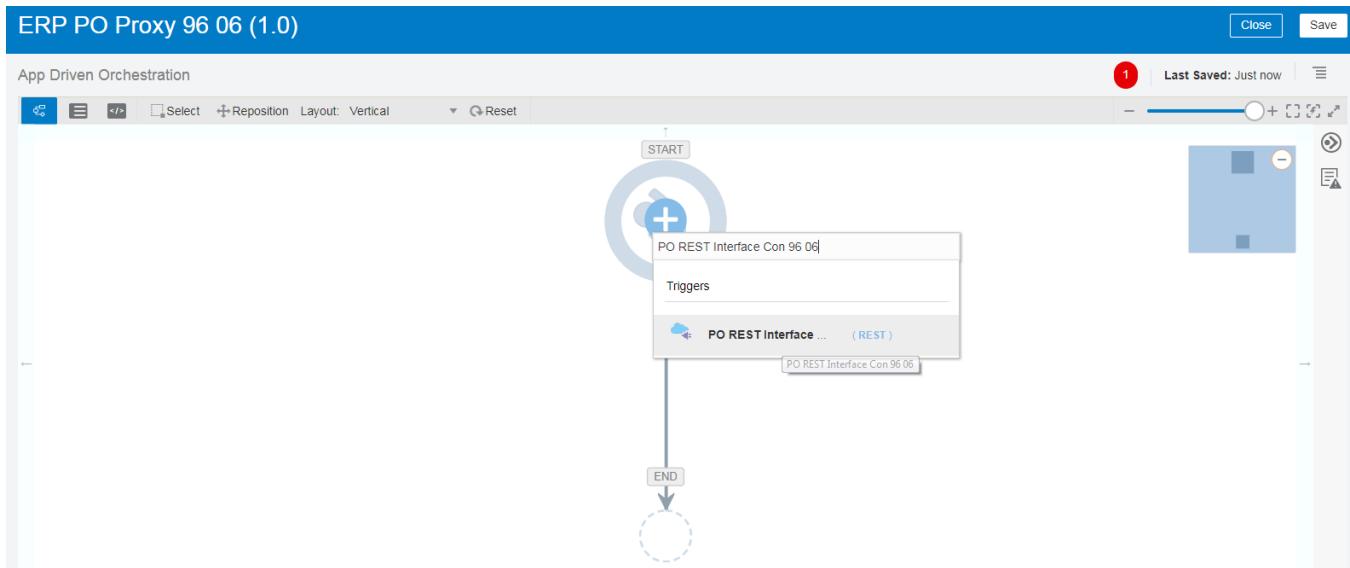
 **Create New Integration**
Enter information that describes this integration.

Describe this integration Use a meaningful name and description that will help others find and understand this integration. The Identifier and Version can be set only when the integration is created. The combination of Identifier and Version must be unique.

* What do you want to call your integration? <input type="text" value="ERP PO Proxy 96 06"/>	What does this integration do? <input type="text" value="This integration is a rest proxy for Purchase order service which will be invoked from VBCS page"/>
* Identifier <input type="text" value="ERP_PO_PROXY_96_06"/>	Which package does this integration belong to? <input type="text" value="Enter a new or existing package name"/>
* Version <input type="text" value="01.00.0000"/>	Create Cancel

Click **Create**. The integration canvas is displayed

5. Hover over the circle next to Start, and click on the + sign
6. Select the REST Connection (PO REST Interface Con 96 06) from the list which is pre created for you
7. Enter details as in the screenshot below to define the endpoint in the flow



8. In the Basic info page provide the endpoint name, uri, HTTP Verb and Select options for Request/Response payload.
Click Next

Basic Info	
Request Parameters	* What do you want to call your endpoint? <input type="text" value="POPROXY"/>
Request	What does this endpoint do? <input type="text" value="REST interface for PO Service"/>
Request Headers	* What is the endpoint's relative resource URI? <input type="text" value="/PO_PROXY"/>
CORS Configuration	* What action do you want to perform on the endpoint? <input type="text" value="POST"/>
Response	
Response Headers	
Summary	

Based on your selections, you can add parameters or configure a request and/or response for this endpoint.

Select any options that you want to configure:

Add and review parameters for this endpoint
 Configure a request payload for this endpoint
 Configure this endpoint to receive the response

9. In the Request page Select Request payload format as “JSON Sample” and provide the sample JSON below by clicking on the inline link

```
{
  "id": 24,
  "IOClId": 3,
  "orderNumber": "163521",
  "pOHeaderId": "300000074157561",
  "procurementBULd": "300000046987012",
```

```

"procurementBusinessUnit": "US1 Business Unit",
"soldToLegalEntity": "US1 Legal Entity",
"supplier": "Dell.",
"supplierId": "300000047414679"
}

```

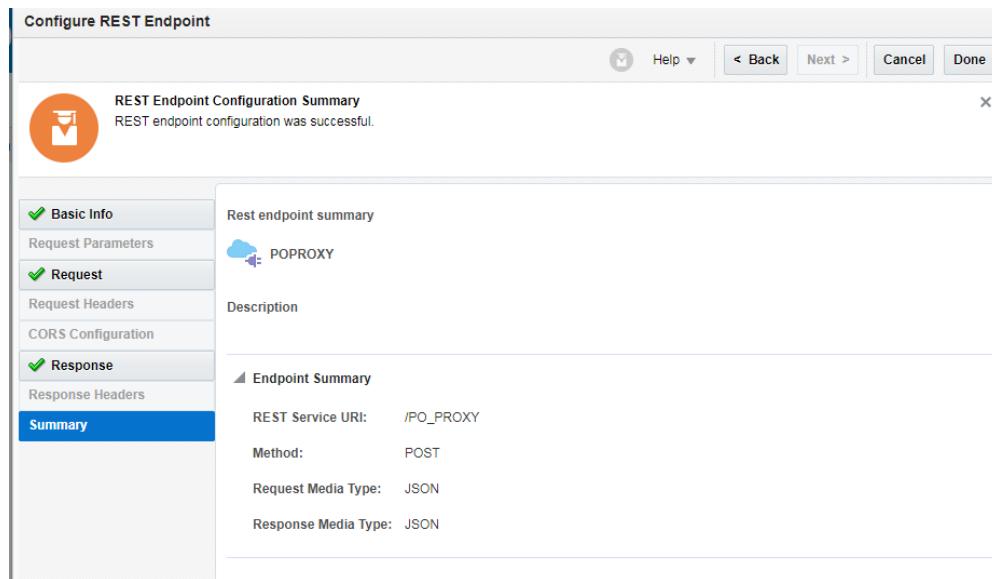
10. In the Response page Select Response payload format as “JSON Sample” and provide the sample JSON below by clicking on the inline link

```

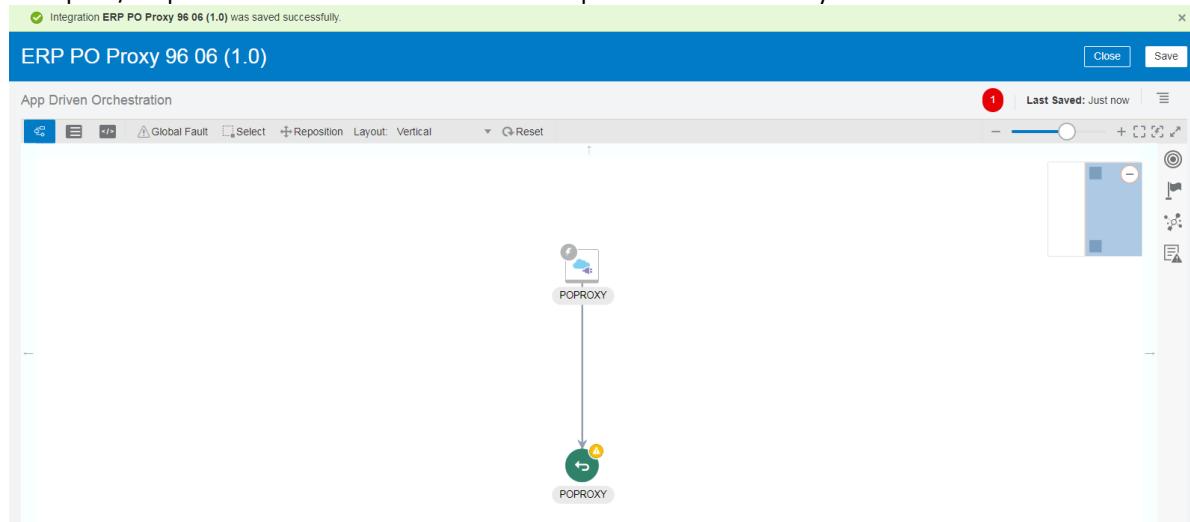
{
"pOHeaderId": "300000074157561",
"orderNumber": "162180",
"soldToLegalEntityId": "300000074157561",
"changeOrderNumber": 3,
"requestStatus": "SUCCESS"
}

```

11. Review the Summary page and click on Done

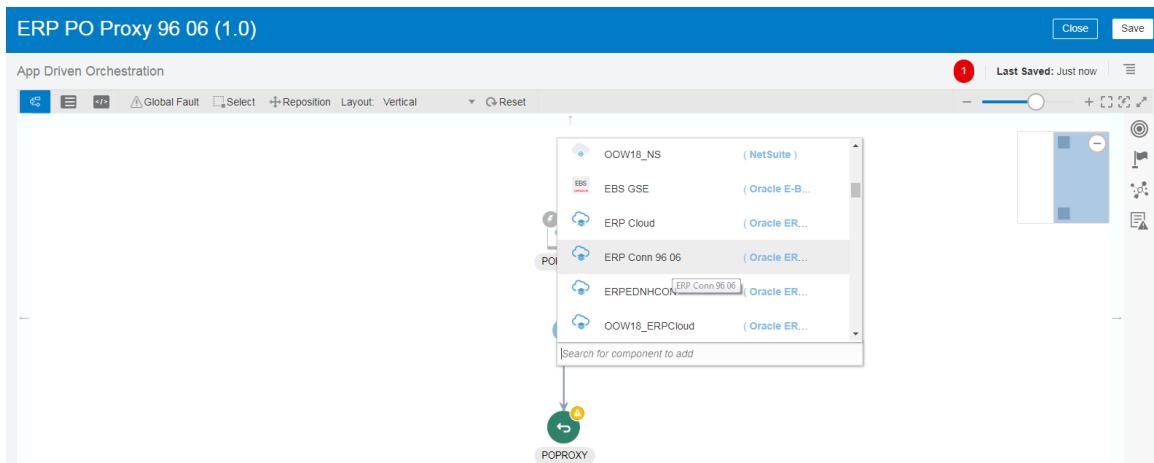


A Request/Response Flow is created. Delete the “Map to POPROXY” activity



Oracle Product Management

12. Hover on the wiring next to POPROXY activity and Click on “+” sign and add the “ERP Conn 96 06” connection which is pre created



13. In the Basic info page provide endpoint name , endpoint description and Click Next

The screenshot shows the 'Configure Oracle ERP Cloud Endpoint' wizard. On the left, a sidebar has 'Basic Info' selected. The main area shows the 'Basic Info' step of the wizard. It asks for the endpoint name ('changePO') and its purpose ('endpoint to update Purchase order'). Buttons for 'Help', '< Back', 'Next >', 'Cancel', and 'Done' are at the top.

14. In the Actions Page, Select the Radio Button “Query, Create, Update or Delete Information” to consume Purchase Order V2 Service

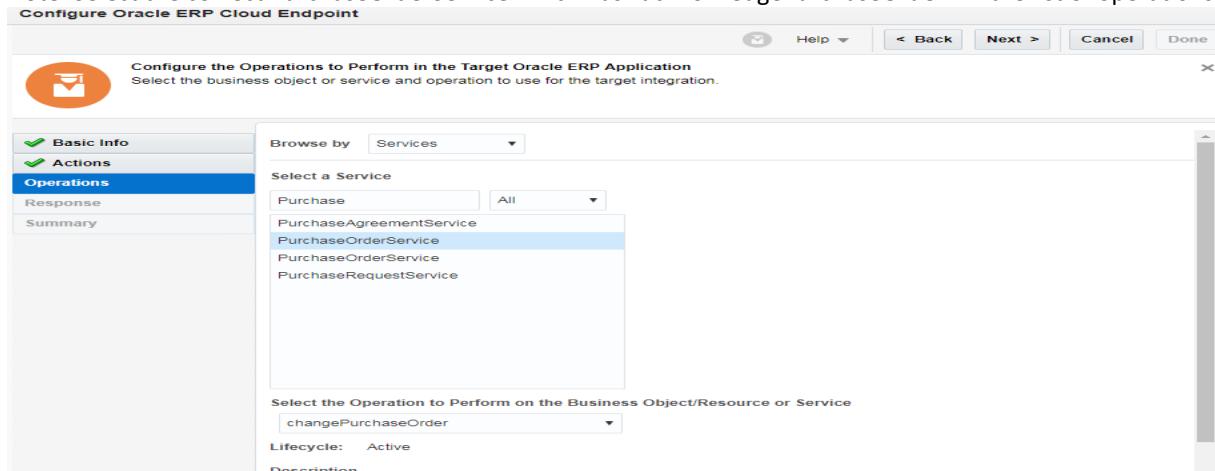
The screenshot shows the 'Configure Oracle ERP Cloud Endpoint' wizard. On the left, a sidebar has 'Actions' selected. The main area shows the 'Actions' step of the wizard. It lists three options: 'Query, Create, Update or Delete Information' (selected and highlighted with a red box), 'Import Bulk Data into Oracle ERP Cloud', and 'Send Files to ERP Cloud'. The 'Query, Create, Update or Delete Information' option is described as performing operations like Find Catalog, Create Orders, Update Accounts, Process Expenses etc. Buttons for 'Help', '< Back', 'Next >', 'Cancel', and 'Done' are at the top.

Oracle Product Management

15. In the Operations Page Select Browse By -> Services and Search for Purchase Order.

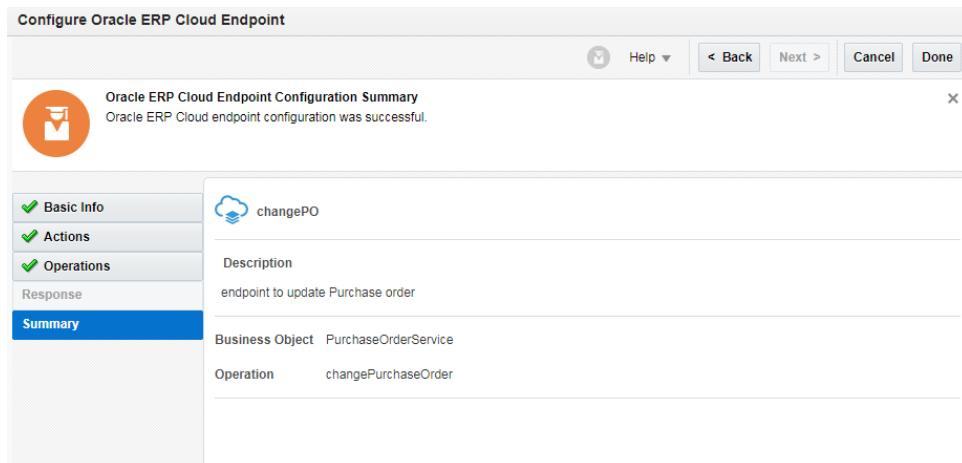
Select PurchaseOrderService and changePurchaseOrder operation to consume in our flow.

Note: Select the correct PurchaseOrderService which has "acknowledgePurchaseOrder" in the list of operations

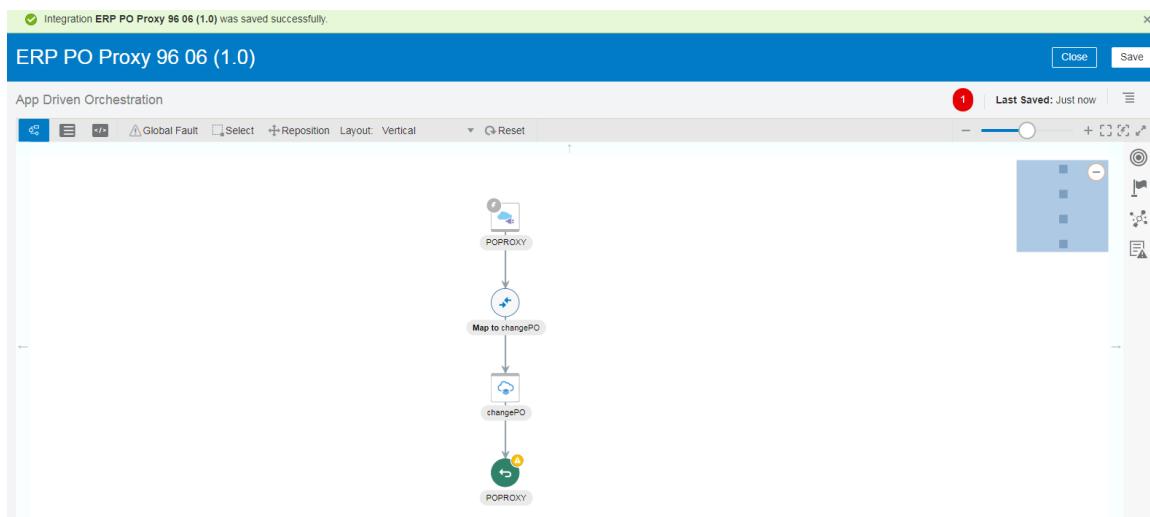


Click Next

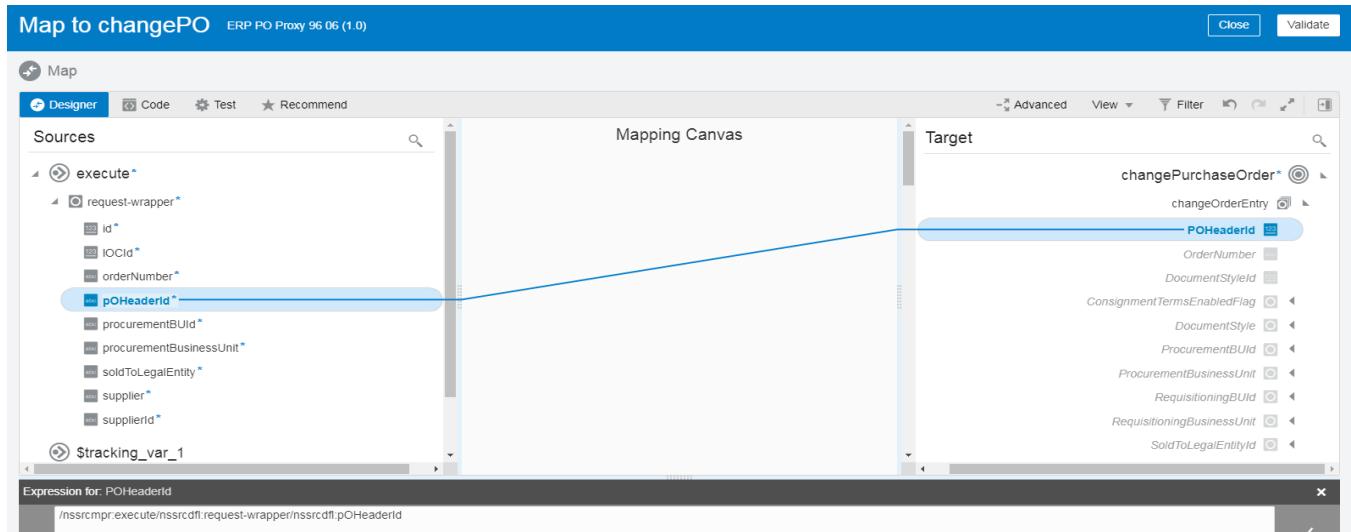
16. Review the Summary Page and Click Done



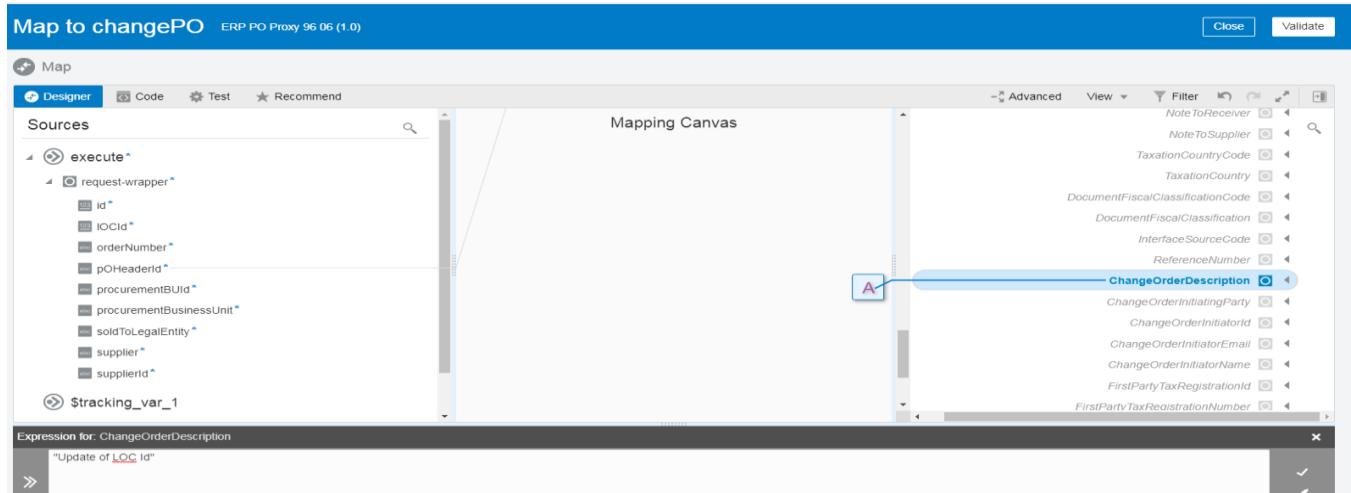
17. An invoke activity is configured with changePO operation and an empty Map (Map to changePO) activity is created



18. Edit the Map (Map to changePO) activity and provide below mapping

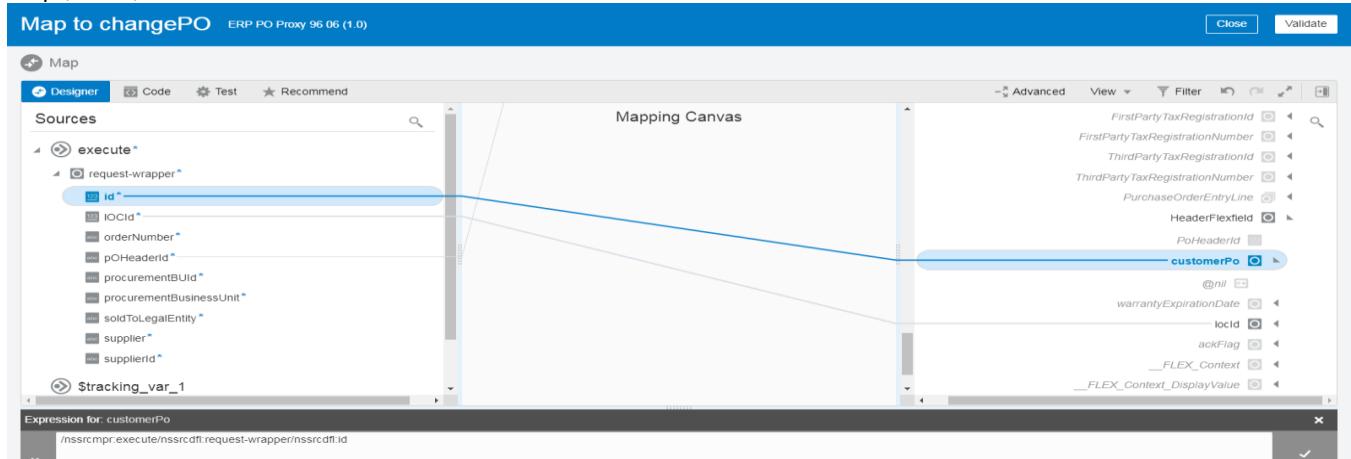


Select the element changeOrderEntry -> ChangeOrderDescription which opens up an expression window. Enter "Update of LOId"

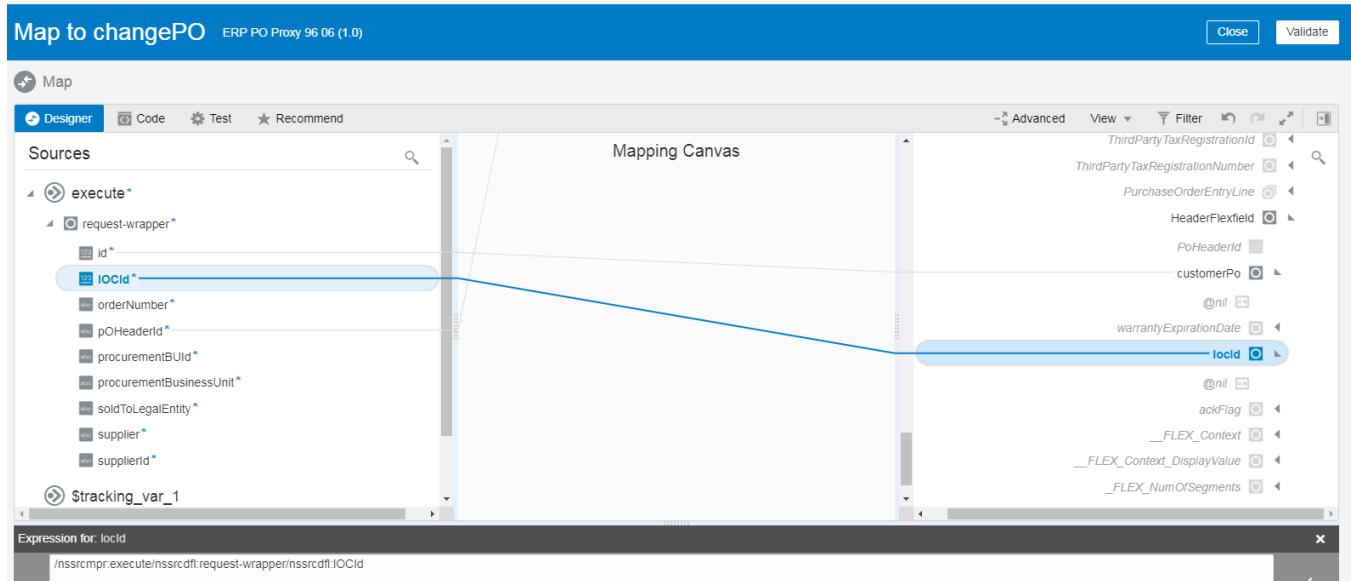


Expand "HeaderFlexField" element and map the following fields from Source to Target

Map \$Id -> \$cusotmerPO

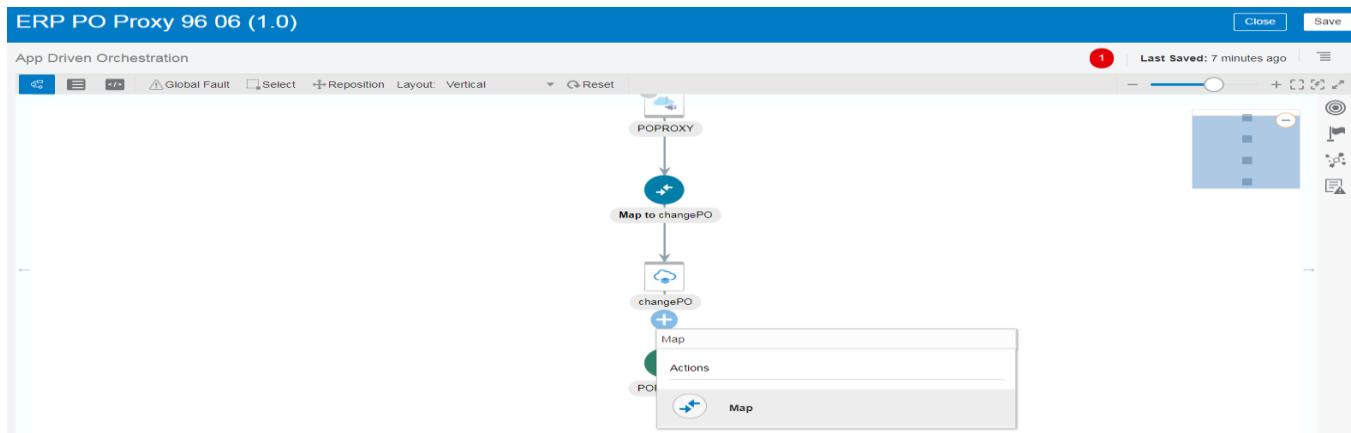


Map IOClId -> locId



Validate and Save your integration

19. Hover on the wiring next to the changePO invoke activity and Click + sign to add a Map activity



20. Edit the “Map to POPROXY” activity and perform the mapping as per below

changePurchaseOrderResponse -> result->POHeaderId **TO** response-wrapper -> poHeaderId
 changePurchaseOrderResponse -> result->OrderNumber **TO** response-wrapper -> orderNumber
 changePurchaseOrderResponse -> result->SoldtoLegalEntityId **TO** response-wrapper -> soldToLegalEntityId
 changePurchaseOrderResponse -> result->ChangeOrderNumber **TO** response-wrapper -> changeOrderNumber
 changePurchaseOrderResponse -> result->RequestStatus **TO** response-wrapper -> requestStatus

The screenshot shows the Oracle Integration Map to POPROXY interface. The Sources pane on the left lists various components and their fields, such as execute, \$changePO, \$tracking_var_1, and \$tracking_var_2. The Target pane on the right shows the target system's structure, including executeResponse, response-wrapper, pOHHeaderId, orderNumber, soldToLegalEntityId, changeOrderNumber, and requestStatus. A central Mapping Canvas displays the flow of data from Sources to Target, with specific fields like OrderNumber being mapped to pOHHeaderId.

Click on Validate and Close

21. Your Final integration flow...

The screenshot shows the final integration flow diagram titled "ERP PO Proxy 96 06 (1.0)". It consists of four main components connected sequentially: POPROXY, Map to changePO, changePO, and another Map to POPROXY. The flow starts with POPROXY, followed by a mapping step, then the changePO component, and finally another mapping step before ending with POPROXY. The interface includes various configuration options and a toolbar at the top.

22. Click on Top right Corner and select Tracking. Add the below Business Identifiers for Tracking your instance

The screenshot shows the "Business Identifiers For Tracking" configuration screen. On the left, a tree view lists business identifiers under the "Source" section, including execute, *execute, *request-wrapper, *id, *IOCId, *orderNumber, *pOHHeaderId, *procurementBUld, *procurementBusinessUnit, *soldToLegalEntity, *supplier, and *supplierId. On the right, a table defines tracking fields:

Primary	Tracking Field	Tracking Name	Tracking Variable	Help Text
<input checked="" type="checkbox"/>	orderNumber	order Number	tracking_var_1	How to track it?
<input checked="" type="checkbox"/>	id	id	tracking_var_2	How to track it?
<input checked="" type="checkbox"/>	IOCId	IOC Id	tracking_var_3	How to track it?

Click on Save and close the integration flow.

Activate Integration (Flow2)

- Click on Activate option for the integration flow "ERP PO Proxy <ClassId><StudentId>" and check Enable Tracing option and check Include Payload option and click on Activate button.

Activate Integration

ERP PO Proxy 96 06 (1.0)

Oracle Recommends

Contribute integration mappings to Oracle Recommends.

i Oracle Integration leverages the collective intelligence to recommend which fields should be mapped while developing an integration. These recommendations are built based on the mappings contributed to Oracle Recommends anonymously. Unselect the checkbox if you do not wish to contribute the mappings. You may change this in recommendations page from settings menu.

Oracle Asserter: When Asserter recording is enabled, payloads will be captured and integration instances will be recorded. Recordings can be played later and maximum five recordings will be maintained for an integration.

Enable Asserter Recording

Tracing: When tracing is enabled, integration activity can be viewed in the Activity Stream.

Enable Tracing

Include Payload

⚠ When payload is included, sensitive information from the payload is written into log files, which can be downloaded and viewed. This may pose a security risk, and also slow down your system. Not recommended in a production environment.

Cancel **Activate**

- Click on "How to run" icon

The screenshot shows a list of integration flows. The second item is "ERP PO Event VBCS 96 06". To its right, there is a "How to run" icon, which is highlighted with a red circle. A tooltip for this icon provides the metadata URL: https://octraining1-oicpm.integration.ocp.oraclecloud.com:443/ic/api/integration/v1/flows/rest/ERP_PO_PROXY_96_06/1.0/metadata.

Open the metadata url generated in a browser window.
This metadata url provides all the description about your REST Endpoint

Endpoint Description**Endpoint URL**

https://OICPMDemoUMic-oicpm.uscom-central-1.oraclecloud.com:443/ic/api/integration/v1/flows/rest/ERP_PO_PROXY_96_0610/PO_PROXY

Swagger

https://OICPMDemoUMic-oicpm.uscom-central-1.oraclecloud.com:443/ic/api/integration/v1/flows/rest/ERP_PO_PROXY_96_0610/metadata/swagger

How to Run

http://www.oracle.com/pls/topic/lookup?ctx=oic_en&id=ICSUG-GUID-205B916C-1075-4603-A9E2-72A6C8C4AB3C

Resource /PO_PROXY**Method POST****Request****Request Media Type**

- application/json

Request sample

```
{ "id": 24, "JOCId": 3, "orderNumber": "163521", "pOHeaderId": "30000074157561", "procurementBUId": "30000046987012", "procurementBusinessUnit": "US1 Business Unit", "soldToLegalEntity": "US1 Legal Entity", "supplier": "Dell.", "supplierId": "30000047414679" }
```

Response**Response Media Type**

- application/json

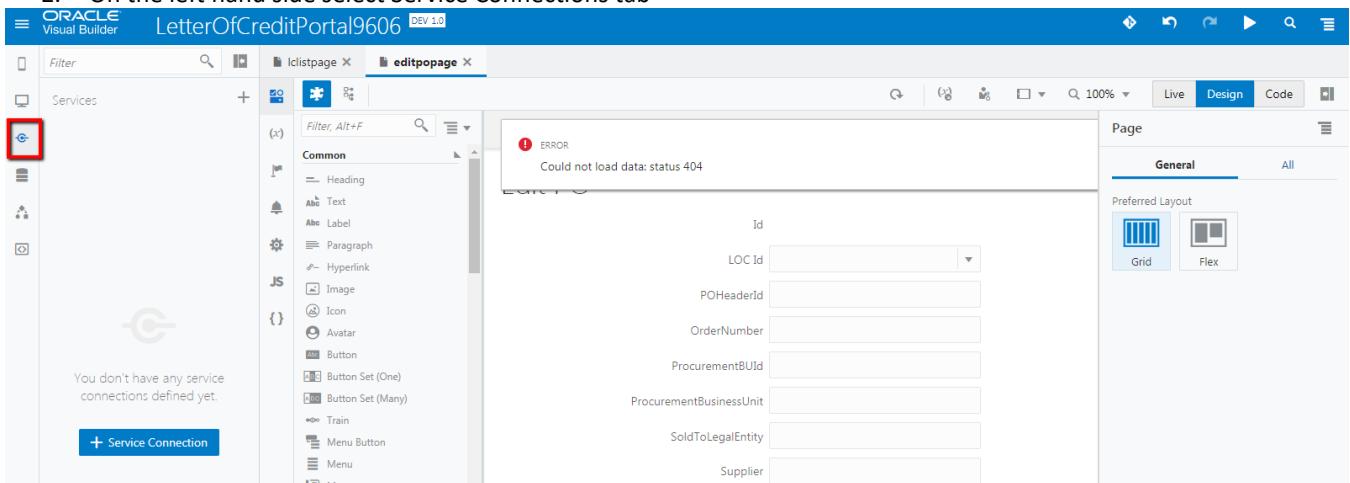
Response sample

```
{ "pOHeaderId": "30000074157561", "orderNumber": "162180", "soldToLegalEntityId": "30000074157561", "changeOrderNumber": 3, "requestStatus": "SUCCESS" }
```

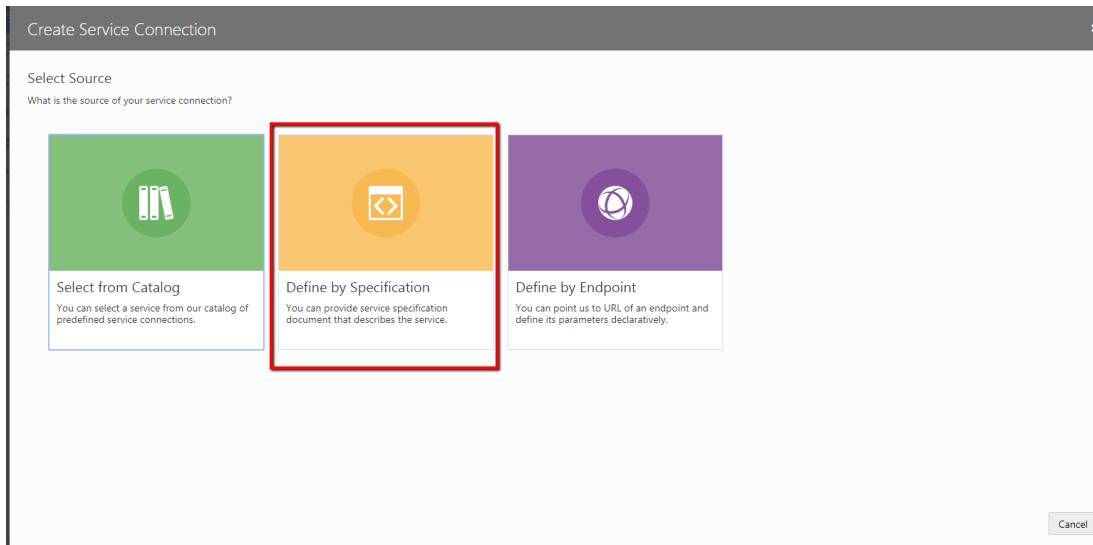
- Copy the Swagger url from the Endpoint description. We will use the swagger definition in next section to create a service connection in VBCS Web application

Invoke ERP Cloud Service from VBCS Web App

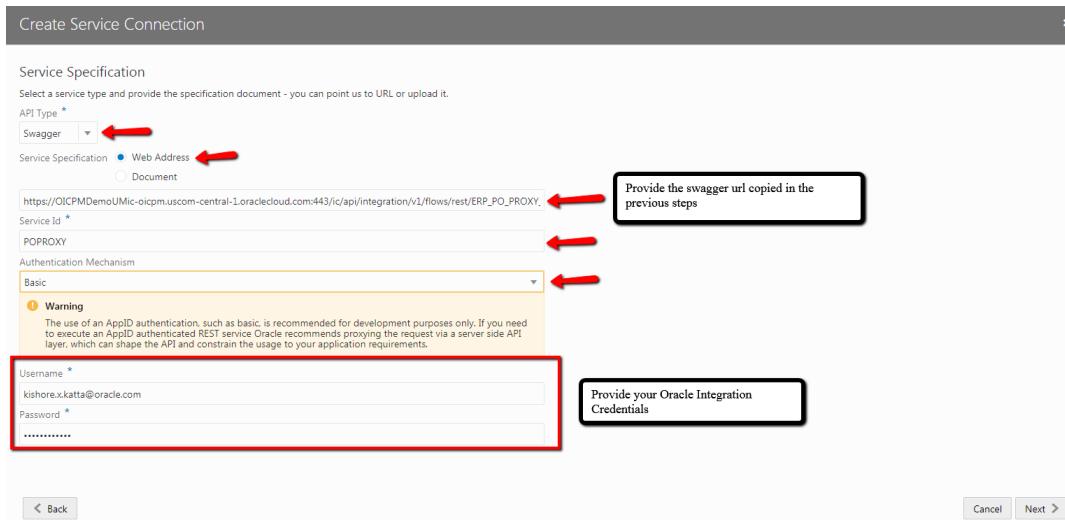
- Select the Visual Builder tab, from the list of applications open the **LetterOfCreditPortal<ClassId><StudentId> Web application**
- On the left hand side select Service Connections tab



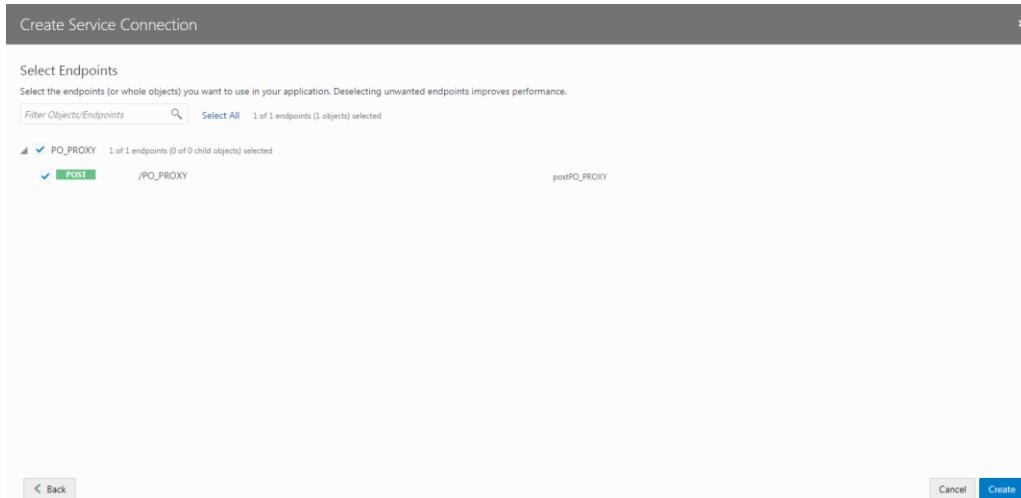
- Click on the “+ Service Connection” to create a new Service Connection to invoke on a button click
- In the “Create Service Connection” page select “Define by Specification”



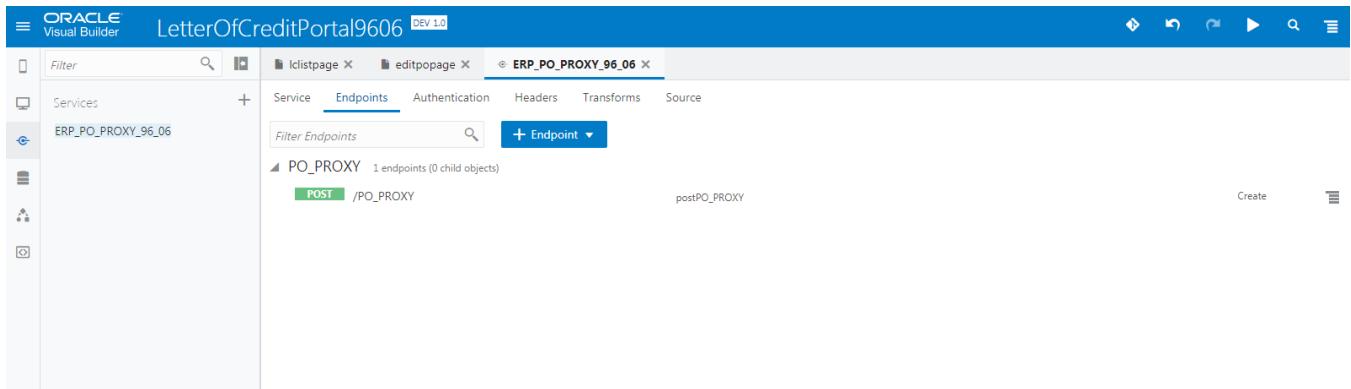
5. Provide the highlighted information as per the screenshot and Click Next



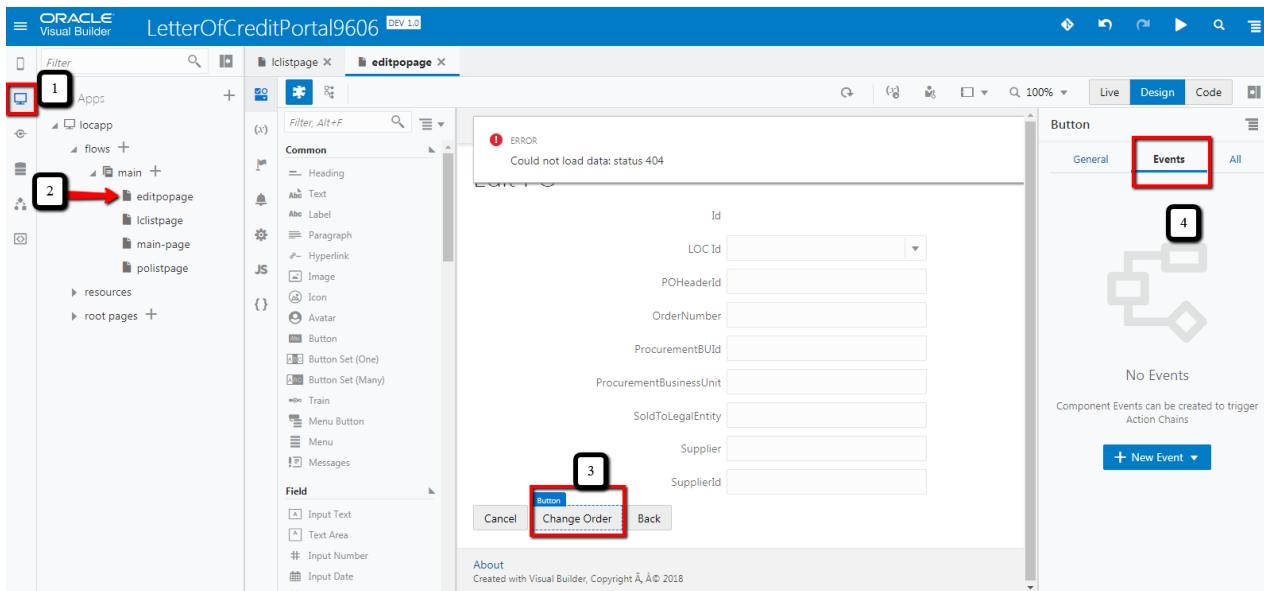
6. In the Select Endpoints page Select the "PO_PROXY" endpoint and click on Create



Connection to the endpoint is created



7. (1) Click on Web application tab and (2) select the “editpoppage” by expanding the localapp->flows->main. (3) Select the “Change Order” button on the web page. (4) On the right hand side select the Events tab



8. Click on New Event and select “Quick Start: ojAction” to create a new Action Chain

The screenshot shows the Oracle Visual Builder interface. On the left, the project tree shows a 'Web Apps' section with a 'locapp' folder containing 'flows' (main, editpage, llistpage, main-page, polistpage) and 'resources'. The main workspace displays a page titled 'editpage' with a form for editing a procurement order. An error message 'Could not load data: status 404' is visible. The Actions palette on the right is open, showing the 'Events' tab with a list of events. A specific event, 'Quick Start: 'ojAction'', is highlighted with a red box.

- Provide Id as "ChangeOrderAction". Drag and drop "Call REST Endpoint" action from Actions palette onto to the "+ Sign"

The screenshot shows the Oracle Visual Builder interface. The Actions palette on the left lists various actions under the 'General' category, including 'Assign Variables', 'Call Action Chain', 'Call Component Method', 'Call Module Function', 'Call REST Endpoint', 'Fire Custom Event', 'Fire Data Provider Event', 'Fire Notification', 'Logout', 'Reset Variables', and 'Transform Chart Data'. The 'Logic' section includes 'For Each', 'If', and 'Return'. On the right, an 'Action Chain' editor is shown with a 'Start' node and a '+' sign for adding more nodes. A red arrow points from the 'Call REST Endpoint' action in the palette towards the '+' sign in the Action Chain editor.

The screenshot shows the Oracle Visual Builder interface. On the left, there's a navigation sidebar with 'Web Apps' expanded, showing 'locapp' with 'flows' containing 'main', which includes 'editpage', 'listpage', 'main-page', and 'polistpage'. Below this are 'resources' and 'root pages'. In the center, the 'Actions' palette is open under the 'General' tab, displaying various actions like 'Assign Variables', 'Call Action Chain', 'Call Component Method', 'Call Module Function', 'Call REST Endpoint', 'Fire Custom Event', 'Fire Data Provider Event', 'Fire Notification', 'Logout', 'Reset Variables', and 'Transform Chart Data'. Below the General tab is a 'Logic' tab with 'For Each', 'If', and 'Return' options. To the right, a BPMN-style diagram shows a 'Start' node connected to a central circle containing a 'C' symbol, which is labeled 'Call REST Endpoint'. A red box highlights this central circle. To the far right, a panel titled 'Call REST Endpoint' displays a placeholder icon and the message 'No endpoint selected. You can call an endpoint of a service connection or business object.' with a 'Select Endpoint' button.

10. Click on "Select Endpoint" and Configure the Service Connection created earlier.

This screenshot shows the 'Select Endpoint' dialog box. At the top, it says 'Select Endpoint'. Below that is a 'Filter' input field and a search icon. The main area contains a tree view of service connections. Under 'Service Connections', 'ERP_PO_PROXY_96_06' is expanded, showing its sub-sections: 'PO_PROXY'. Below this, 'POST /PO_PROXY' is highlighted in green. Other sections include 'Business Objects' and 'Process Objects'. At the bottom right of the dialog are 'Cancel' and 'Select' buttons.

Click on Select. The endpoint is now configured to be invoked from the action chain.

11. Select “Parameters” -> body which opens a mapper interface page

Call REST Endpoint: callRestEndpoint1

Sources

- Action Chain
- Results
- Page
 - IOCLListSDP
 - pOld
 - {}** pRecord
 - A pRecordETag
- System
- Flow
- System
- Application

Target

Parameters

- {} body
- A filePath
- {} requestTransformOptions

Expression for: body

\$poRecord

Cancel Save

12. We want to Map the “editpopage” page variables to the endpoint interface.

Map \$poRecord(Sources) TO body (Target)

Call REST Endpoint: callRestEndpoint1

Sources

- Action Chain
- Results
- Page
 - IOCLListSDP
 - pOld
 - {}** pRecord
 - A pRecordETag
- System
- Flow
- System
- Application

Target

Parameters

- {}** body
- A filePath
- {} requestTransformOptions

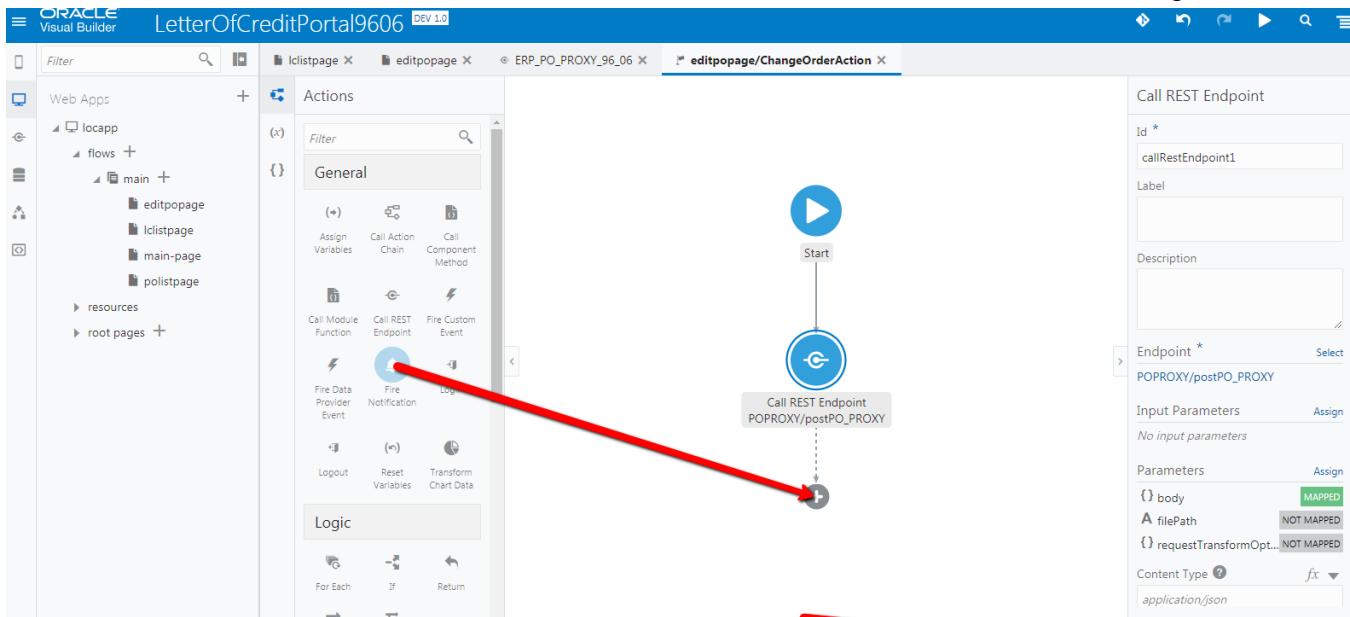
Expression for: body

\$poRecord

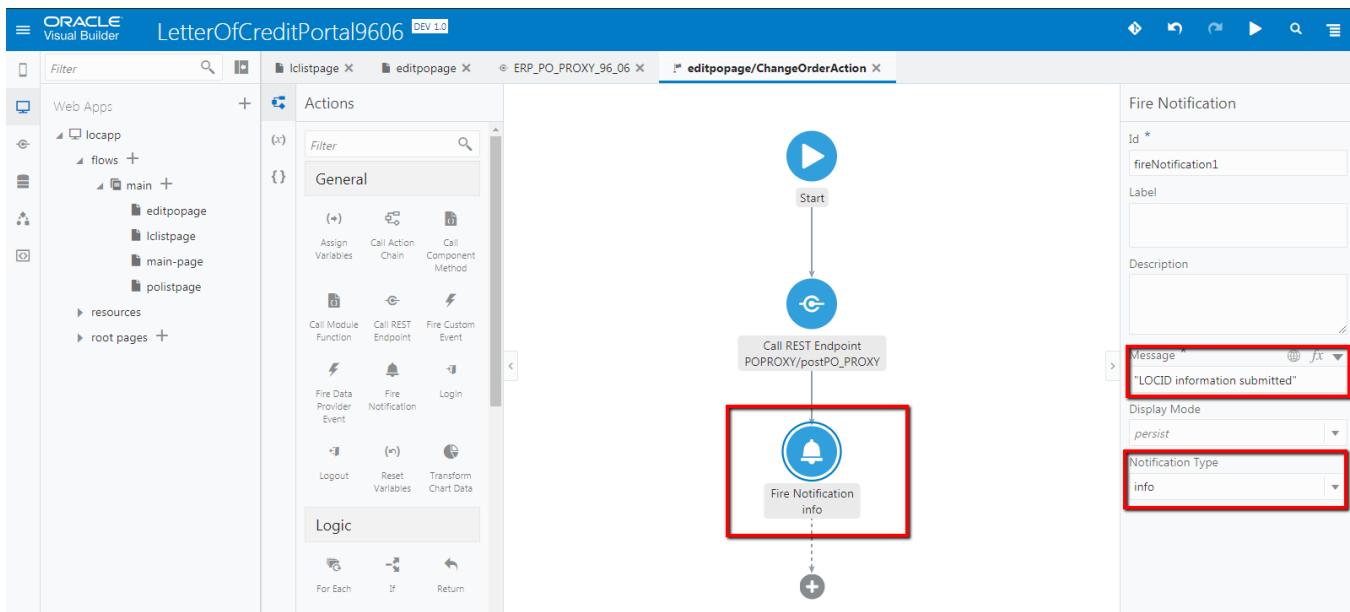
Cancel Save

An expression is generated in the expression window based on the mapping. poRecord is a page variable which holds the data from the “editpopage”. Click on Save. You should see the body parameter now marked as “**MAPPED**”

13. From the Action palette Select “Fire Notification” action on to the “+” sign next to the “Call Rest Endpoint” activity

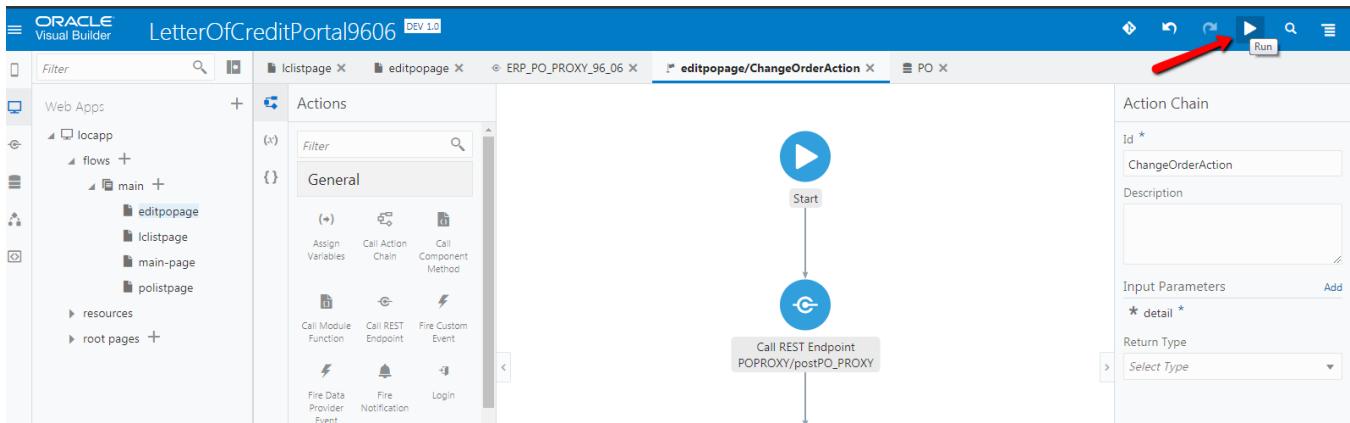


14. Change the Notification type -> info and provide a Message in the text area



Running the VBCS Web App

1. Click on Run



2. The Web application opens up in a separate browser window. We have given a basic web app for you to test the end to end flow

The screenshot shows a web browser window with the title 'LC Portal'. The header includes the text 'LC Portal' and the email 'kishore.x.katta@oracle.com'. Below the header are two buttons: 'LC List Page' and 'PO List Page'. The main content area is currently empty, indicating the user has just opened the page.

3. Click on LC List Page which displays all the Letter of Credits Available in a Custom table. Go back to the main page

The screenshot shows a web browser window with the title 'LC Portal'. The header includes the text 'LC Portal' and the email 'kishore.x.katta@oracle.com'. Below the header is a table displaying LOC records:

LOCId	LOCAmount	LOCBank	LOCExpiry	LOCOpeningDate	LOCStatus	LOCType
1	10000	ABC Bank	2020-01-01	2019-01-01	Active	Type1
2	20000	XYZ Bank	2020-01-01	2019-01-01	Active	Type2
3	25000	LVB Bank	2020-01-01	2019-01-01	Active	Type1
4	30000	OCOCO Bank	2020-01-01	2019-01-01	Active	Type2
5	40000	Green Bank	2020-01-01	2019-01-01	Active	Type3
6	50000	Others Bank	2020-01-01	2019-01-01	Active	Type4

A 'Back' button is visible at the bottom left of the page.

4. Click on PO List Page . All the PO records that was created earlier in ERP Cloud was synced into VBCS

The screenshot shows a web browser window with the title 'LC Portal'. The header includes the text 'LC Portal' and the email 'kishore.x.katta@oracle.com'. Below the header is a table displaying PO records:

POHeaderId	300000176127931
OrderNumber	163531
Supplier	Dell Inc.
LOCId	

Buttons for 'Edit PO' and 'Back' are visible at the top left of the page.

5. Select PO that you created and Click on “Edit PO”

LC Portal

Edit PO Back

POHeaderId	300000176127931
OrderNumber	163531
Supplier	Dell Inc.
LOCId	

6. In the “Edit PO” page provide “LOCId” information. Leave the rest of the parameters intact

LC Portal kishore.x.katta@oracle.com

Edit PO

Id 63

LOC Id	4
POHeaderId	300000176127931
OrderNumber	163531
ProcurementBUID	300000046987012
ProcurementBusinessUnit	US1 Business Unit
SoldToLegalEntity	US1 Legal Entity
Supplier	Dell Inc.
SupplierId	300000047414679

Cancel Change Order Back

7. Click on “Change Order” button. Which invokes your “ERP PO Proxy <ClassId> <StudentId>” Integration flow

8. You should see an instance is created. Go to Integrations -> Monitoring -> Tracking

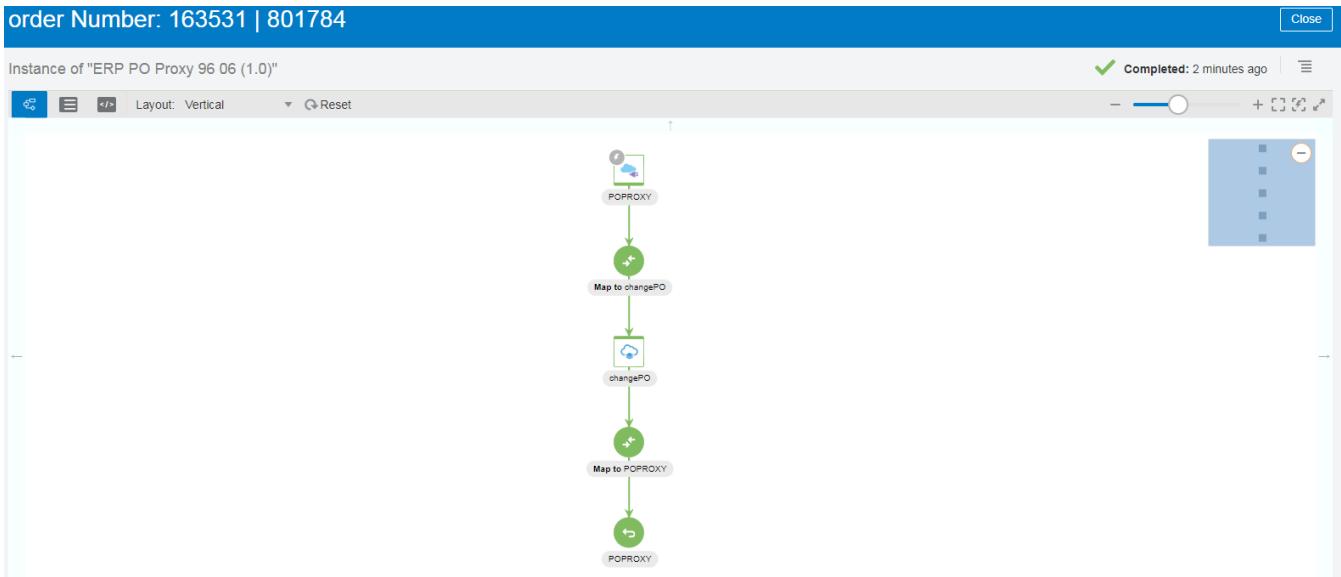
Monitoring ORACLE Integration Cloud

Track Instances

Jan 26, 2019 9:59:09 AM UTC Last 1 Hour

	order Number: 163531	Instance ID: 801784	id 63	Started just now
	ERP PO Proxy 96 06 1.0	LOC Id 4	Completed just now	Duration 33 sec
	COMPLETED			

9. Select the Order Number and it confirm that the flow is completed and green



10. Also, observe a notification created in the “Edit PO Page”. Confirming the payload is submitted to the Proxy Integration flow

The screenshot shows the Oracle ERP Cloud 'Edit Order' page. A red box highlights a success message: "INFORMATION: LOCID information submitted". Below the message is a form with fields: Id (63), LOCId (4), POHeaderId (300000176127931), OrderNumber (163531), ProcurementBUID (300000046987012), ProcurementBusinessUnit (US1 Business Unit), SoldToLegalEntity (US1 Legal Entity), Supplier (Dell Inc.), and SupplierId (300000047414679). At the bottom are 'Cancel', 'Change Order', and 'Back' buttons.

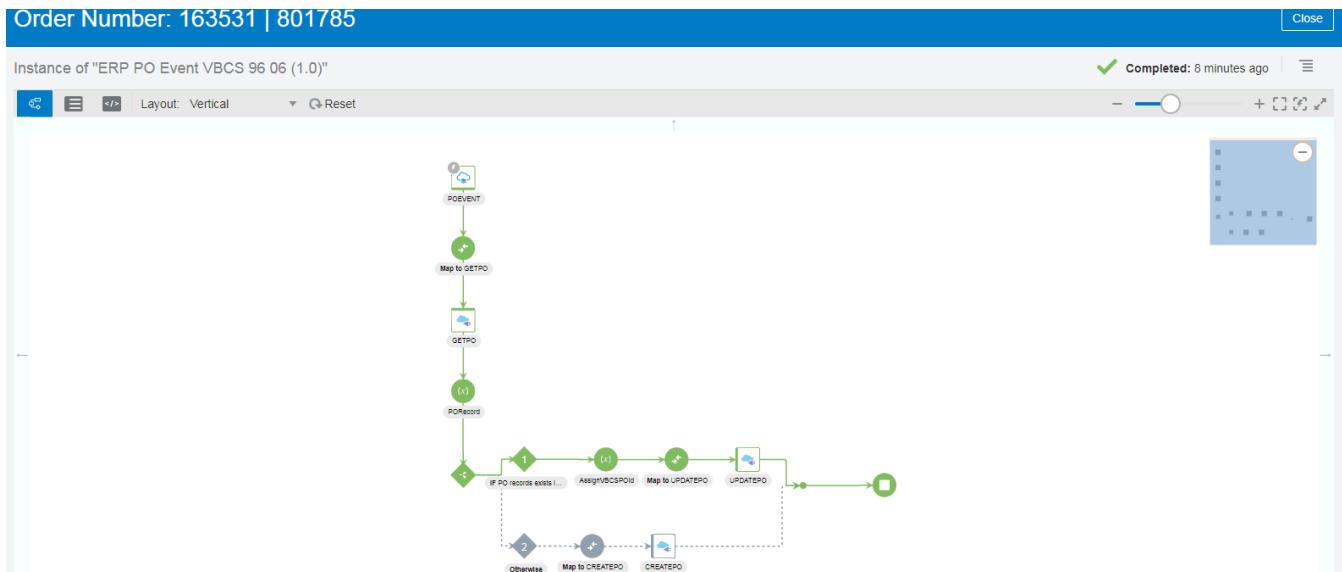
11. In the Integrations -> Monitoring -> tracking page, you can observe another instance got created. As, LOCID is got updated in ERP Cloud Flow 1 is triggered again.

Note: Purchase Order Event is raised both for Create and Update records

The screenshot shows the Oracle Integration Cloud 'Track Instances' page. It displays two instances: 'Flow1' and 'Flow2'. Both instances have an 'Order Number: 163531' and are marked as 'COMPLETED'. The 'Flow1' instance has an 'Instance ID: 801785' and 'PO Header Id: 300000176127931'. The 'Flow2' instance has an 'Instance ID: 801784' and 'id: 63'. Both instances show a duration of 02 sec and 33 sec respectively. The left sidebar shows 'Tracking' is selected.

Open the Instance and we see that “If” path is evaluated to Update the PO into VBCS

Integrating Your ERP with Oracle Integration



12. Now go back to the PO List page. It should now show the “LOCIId” updated.

LC Portal

Edit PO Back

POHeaderId	300000176127931
OrderNumber	163531
Supplier	Dell Inc.
LOCIId	4

Congratulations, you have now completed Flow 2 and an end-to-end Flow to capture PO Events and update with cross reference of Letter of Credit Identifier from VBCS Web app.

